Operating Instructions Manual

netGateway DTM for netTAP and netBRICK

Configuration of Gateway Devices

Hilscher Gesellschaft für Systemautomation mbH

www.hilscher.com

DOC081201OI02EN | Revision 2 | English | 2010-02 | Released | Public

Table of Contents

1 INTRODUCTION 5
1.1 About this Manual 5
1.1.1 Online Help 6
1.1.2 List of Revisions 6
1.1.3 Conventions in this Manual 7
1.2 Legal Notes 8
1.2.1 Copyright 8
1.2.2 Important Notes 8
1.2.3 Exclusion of Liability 9
1.2.4 Warranty 9
1.2.5 Export Regulations 10
1.2.6 Registered Trademarks 10
1.3 About netGateway DTM 11
1.3.1 Requirements 11
1.4 Dialog Structure of the netGateway DTM 12
1.4.1 General Device Information 13
1.4.2 Navigation Area 13
1.4.3 Dialog Panes 14
1.4.4 Controls and Dialog Buttons 15
1.4.5 Status Bar 16

2 CONFIGURATION OF NT 100 AS GATEWAY 17

3 SETTINGS 29
3.1 Overview Settings 29
3.2 Driver 30
3.2.1 Selecting the Driver 30
3.3 Device Assignment 31
3.3.1 Scanning for Devices 32
3.3.2 Selecting the Device 34

4 CONFIGURATION 35
4.1 Overview Configuration 35
4.2 Settings 36
4.3 Licensing 37
4.3.1 Which licenses are present in the Device? 38
4.3.2 How to order a license 39
4.3.3 How to get the License and transfer it to the Device 44
4.4 Signal Mapping 45
4.4.1 Status Information 48
4.5 Memory Card Management 52
4.6 Configuration EtherCAT Master 53
4.7 Configuration EtherCAT Slave 54
4.7.1 EtherCAT Slave Parameter 55
4.7.2 Settings at the used EtherCAT Master 56
4.8 Configuration EtherNet/IP Scanner 57

© Hilscher, 2009-2010

4.9 Configuration EtherNet/IP Adapter 58
4.9.1 EtherNet/IP Adapter Parameter 60
4.9.2 Settings at the used EtherNet/IP Scanner 62
4.10 Configuration Open Modbus/TCP 63
4.10.1 Open Modbus/TCP Parameter 65
4.10.2 Command Table Open Modbus/TCP 67
4.10.3 Open Modbus/TCP Server 73
4.10.4 Settings at the used Open Modbus/TCP Client 73
4.10.5 Modbus References 74
4.11 Configuration POWERLINK Controlled Node 75
4.11.1 POWERLINK Controlled Node Parameter 77
4.11.2 Settings at the used POWERLINK Managing Node 79
4.12 Configuration PROFINET IO Controller 80
4.13 Configuration PROFINET IO Device (Gateway) 81
4.13.1 PROFINET IO Device Parameter 83
4.13.2 Settings at the used PROFINET IO Controller 84
4.14 Configuration SERCOS III Slave 85
4.14.1 SERCOS III Slave Parameter 86
4.14.2 Settings at the used SERCOS III Master 87
4.15 Configuration CANopen Master 88
4.16 Configuration CANopen Slave 89
4.16.1 CANopen Slave Parameter 91
4.16.2 Settings at the used CANopen Master 93
4.17 Configuration CC-Link Slave 94
4.17.1 CC-Link Slave Parameter 96
4.17.2 Settings at the used CC-Link Master 98
4.18 Configuration DeviceNet Master 99
4.19 Configuration DeviceNet Slave 100
4.19.1 DeviceNet Slave Parameter 102
4.19.2 Settings at the used DeviceNet Master 104
4.20 Configuration PROFIBUS-DP Master 105
4.21 Configuration PROFIBUS-DP Slave 106
4.21.1 PROFIBUS-DP Slave Parameter 108
4.21.2 Settings at the used PROFIBUS-DP Master 110
4.22 Configuration Modbus RTU Master/Slave 111
4.22.1 Modbus RTU Parameter 113
4.22.2 Modbus RTU Master 114
4.22.3 Modbus RTU Slave 121
4.22.4 Modbus References 122
4.23 Configuration ASCII 123
4.23.1 ASCII Settings 125
4.23.2 ASCII Parameters 125
4.23.3 Settings for the ASCII Remote Device 137
4.24 Configuration ‘Serial with netSCRIPT’ 138
4.24.1 netSCRIPT Settings 139
4.24.2 Settings for the ASCII Remote Device 139
4.25 Signal Configuration 140

© Hilscher, 2009-2010

5 DIAGNOSIS 141
5.1 Overview Diagnosis 141
5.2 General Diagnosis 142
5.3 Firmware Diagnosis 145

6 ONLINE FUNCTIONS 146
6.1 Connecting/Disconnecting Device 146
6.2 Download Configuration 148

7 ERROR NUMBERS 149
7.1 Error Code Definition 149
7.2 Overview Error Numbers 150
7.3 General Hardware Error Codes 151
7.3.1 RCX Errors - MID_SYS Task Error Codes 151
7.3.2 RCX Errors - Common Status Codes 153
7.3.3 RCX Errors - Status & Error Codes 154
7.4 ODM Error Codes 155
7.4.1 General ODM Error Codes 155
7.4.2 General ODM Driver Error Codes 156
7.4.3 ODM cifX Driver Specific Errors 157
7.5 ODM DBM V4 Error Codes 161

8 LISTS 167
8.1 List of Figures 167
8.2 List of Tables 167

9 GLOSSARY 169

10 APPENDIX 170
10.1 User Rights 170
10.1.1 Settings 170
10.1.2 Configuration 170
10.2 References 171
10.3 Contacts 172

© Hilscher, 2009-2010

1 Introduction

1.1 About this Manual

This manual provides information on how to set and configure the device parameters of a netTAP NT 100 device respectively a netBRICK NB 100 device using the netGateway DTM, and what can be read from the diagnosis panes.

Dialog Panes

The table below gives an overview for the individual dialog panes descriptions:

Section	Subsection	Page
Settings	*Overview Settings*	29
	Driver	30
	Device Assignment	31
Configuration	*Overview Configuration*	35
	Settings	*36*
	Licensing	*37*
	Signal Mapping	*45*
	Memory Card Management	*52*
Diagnosis	*Overview Diagnosis*	*141*
	General Diagnosis	*142*
	Firmware Diagnosis	*145*

Table 1: Descriptions Dialog Panes

© Hilscher, 2009-2010

1.1.1 Online Help

The netGateway DTM contains an integrated online help facility.

- To open the online help, click on the **Help** button or press the **F1** key.

1.1.2 List of Revisions

Index	Date	Version	Component	Chapter	Revision
1	2008-12-08	1.0.0.x 1.0.0.x	netGatewayDTM.dll netGatewayGUI.ocx	all	created
2	2010-02-19	1.200.x.x	netGatewayDTM.dll netGatewayGUI.ocx		Expanded to 19 protocols

© Hilscher, 2009-2010

1.1.3 Conventions in this Manual

Operation instructions, a result of an operation step or notes are marked as follows:

Operation Instructions:

➢ <instruction>

Or

1. <instruction>
2. <instruction>

Results:

➯ <result>

Notes:

Important: <important note>

Note: <note>

<note, were to find further information>

© Hilscher, 2009-2010

1.2 Legal Notes

1.2.1 Copyright

© 2008 Hilscher Gesellschaft für Systemautomation mbH

All rights reserved.

The images, photographs and texts in the accompanying material (user manual, accompanying texts, documentation, etc.) are protected by German and international copyright law as well as international trade and protection provisions. You are not authorized to duplicate these in whole or in part using technical or mechanical methods (printing, photocopying or other methods), to manipulate or transfer using electronic systems without prior written consent. You are not permitted to make changes to copyright notices, markings, trademarks or ownership declarations. The included diagrams do not take the patent situation into account. The company names and product descriptions included in this document may be trademarks or brands of the respective owners and may be trademarked or patented. Any form of further use requires the explicit consent of the respective rights owner.

1.2.2 Important Notes

The user manual, accompanying texts and the documentation were created for the use of the products by qualified experts, however, errors cannot be ruled out. For this reason, no guarantee can be made and neither juristic responsibility for erroneous information nor any liability can be assumed. Descriptions, accompanying texts and documentation included in the user manual do not present a guarantee nor any information about proper use as stipulated in the contract or a warranted feature. It cannot be ruled out that the user manual, the accompanying texts and the documentation do not correspond exactly to the described features, standards or other data of the delivered product. No warranty or guarantee regarding the correctness or accuracy of the information is assumed.

We reserve the right to change our products and their specification as well as related user manuals, accompanying texts and documentation at all times and without advance notice, without obligation to report the change. Changes will be included in future manuals and do not constitute any obligations. There is no entitlement to revisions of delivered documents. The manual delivered with the product applies.

Hilscher Gesellschaft für Systemautomation mbH is not liable under any circumstances for direct, indirect, incidental or follow-on damage or loss of earnings resulting from the use of the information contained in this publication.

1.2.3 Exclusion of Liability

The software was produced and tested with utmost care by Hilscher Gesellschaft für Systemautomation mbH and is made available as is. No warranty can be assumed for the performance and flawlessness of the software for all usage conditions and cases and for the results produced when utilized by the user. Liability for any damages that may result from the use of the hardware or software or related documents, is limited to cases of intent or grossly negligent violation of significant contractual obligations. Indemnity claims for the violation of significant contractual obligations are limited to damages that are foreseeable and typical for this type of contract.

It is strictly prohibited to use the software in the following areas:

- for military purposes or in weapon systems;
- for the design, construction, maintenance or operation of nuclear facilities;
- in air traffic control systems, air traffic or air traffic communication systems;
- in life support systems;
- in systems in which failures in the software could lead to personal injury or injuries leading to death.

We inform you that the software was not developed for use in dangerous environments requiring fail-proof control mechanisms. Use of the software in such an environment occurs at your own risk. No liability is assumed for damages or losses due to unauthorized use.

1.2.4 Warranty

Although the hardware and software was developed with utmost care and tested intensively, Hilscher Gesellschaft für Systemautomation mbH does not guarantee its suitability for any purpose not confirmed in writing. It cannot be guaranteed that the hardware and software will meet your requirements, that the use of the software operates without interruption and that the software is free of errors. No guarantee is made regarding infringements, violations of patents, rights of ownership or the freedom from interference by third parties. No additional guarantees or assurances are made regarding marketability, freedom of defect of title, integration or usability for certain purposes unless they are required in accordance with the law and cannot be limited. Warranty claims are limited to the right to claim rectification.

© Hilscher, 2009-2010

1.2.5 Export Regulations

The delivered product (including the technical data) is subject to export or import laws as well as the associated regulations of different counters, in particular those of Germany and the USA. The software may not be exported to countries where this is prohibited by the United States Export Administration Act and its additional provisions. You are obligated to comply with the regulations at your personal responsibility. We wish to inform you that you may require permission from state authorities to export, re-export or import the product.

1.2.6 Registered Trademarks

Windows® 2000/Windows® XP are registered trademarks of Microsoft Corporation.

All other mentioned trademarks are property of their respective legal owners.

© Hilscher, 2009-2010

1.3 About netGateway DTM

You can use the netGateway DTM to configure a netTAP NT 100 device respectively a netBRICK NB 100 device within a FDT Framework.

1.3.1 Requirements

System Requirements

- Windows® 2000/ Windows® XP
- DVD ROM drive
- Graphic resolution: min. 1024 x 768 pixel
- Keyboard and Mouse

Requirements netGateway DTM

To configure a netTAP NT 100 device or a netBRICK NB 100 device with a DTM the following requirements have to be accomplished:

- Completed hardware installation of a netTAP NT 100 device respectively a netBRICK NB 100 device, where at least a base firmware has to be loaded in the device
- Installed FDT/DTM V 1.2 compliant frame application
- Loaded DTM in the Device Catalog of the FTD Framework

© Hilscher, 2009-2010

1.4 Dialog Structure of the netGateway DTM

The graphical user interface of the DTM is composed of different areas and elements listed hereafter:

1. A header area containing the **General Device Information**,
2. The **Navigation Area** (area on the left side),
3. The **Dialog Pane** (main area on the right side),
4. The general buttons **OK**, **Cancel**, **Apply**, **Help**,
5. The **Status Line** containing information e. g. the online-state of the DTM.

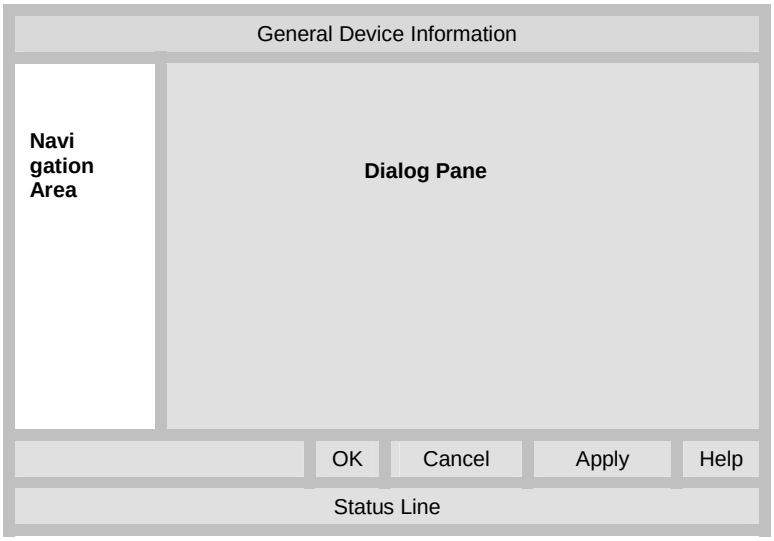


Figure 1: Dialog Structure of netGateway DTM

© Hilscher, 2009-2010

1.4.1 General Device Information

Parameter	Meaning
IO Device	Name of the device
Vendor	Vendor name of the device
Device ID	Identification number of the device
Vendor ID	Identification number of the vendor

Table 2: General Device Information

1.4.2 Navigation Area

The **Navigation Area** at the left side of the dialog provides a tree structure to navigate through the panes of the DTM.

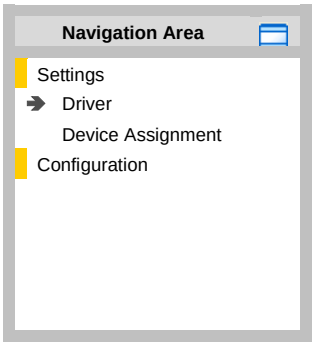


Figure 2: Navigation Area

➢ To access a DTM pane select the respective item of the navigation tree structure.

The **Navigation Area** can be hidden or it can be displayed again.

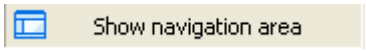
Control	Meaning
	Window button to hide the navigation area, (at the right side of the navigation title bar).
	**Show navigation area** button to open the navigation area, (at the lower left corner of the dialog pane).

Table 3: Hide / display Navigation

© Hilscher, 2009-2010

1.4.3 Dialog Panes

At the dialog pane the **Settings**, **Configuration**, **Diagnosis/Extended Diagnosis** or the **Additional Tools** panes are opened via the corresponding folder in the navigation area.

Settings	
Driver	On the pane **Driver** you can select a driver from the driver list. For further information, refer to section *Driver* on page *30*.
Device Assignment	On the pane **Device Assignment** you must select the device and assign the device to the driver. For further information, refer to section *Device Assignment* on page *31*.
Configuration	
Setting	The **Settings** pane serves for selection of the protocols for the Gateway respectively the Proxyfunctionality. For further information, refer to section *Settings* on page *36.*
Licensing	On the pane **Licensing** you can verify which licenses are available in the netTAP NT 100 device, order licenses and transfer license files into the netTAP NT 100 device. For further information, refer to section *Licensing* on page 37.
Signal Mapping	On the pane **Signal Mapping** you can map the signals (I/O Data) of port X2 to the signals (I/O data) of port X3. For further information, refer to section *Signal Mapping* on page *45.*
Memory Card Management	On the pane **Memory Card Management** you can backup and resotore the firmware and the configuration to respectively from an MMC card. For further information, refer to section *Memory Card* Management on page *52.*
Diagnosis	
Diagnosis	At the **Diagnosis** panes diagnosis information can be read. For further information, refer to section *Overview Diagnosis* on page 141.

Table 4: Overview Dialog Panes

Note: Accessing the **Diagnosis** panes of the netGateway DTM requires an online connection from the netGateway DTM to the netTAP NT 100 device respectively to the netBRICK NB 100 device.

For further information, refer to section *Connecting/Disconnecting Device* on page 146.

© Hilscher, 2009-2010

1.4.4 Controls and Dialog Buttons

In this section, you will find general information on controls and buttons.

1.4.4.1 General Buttons

The table below explains the general buttons in the DTM user interface.

Button	Meaning
OK	To confirm your latest settings, click on the **OK** button. All changed values will be applied on the frame application database. *The dialog then closes.*
Cancel	To cancel your latest changes, click on the **Cancel** button. Answer to the safety query **Configuration data has been changed. Do you want to save the data?** by **Yes**, **No** or **Cancel**. **Yes**: The changes are saved or the changed values are applied on the frame application database. *The dialog then closes.* **No**: The changes are not saved or the changed values are not applied on the frame application database. *The dialog then closes.* **Cancel**: *Back to the DTM.*
Apply	To confirm your latest settings, click on the **Apply** button. All changed values will be applied on the frame application database. *The dialog remains opened.*
Help	To open the DTM online help, click on the **Help** button.

Table 5: General Buttons

1.4.4.2 Table Line Buttons

- To select, create or delete a line in a table use the buttons shown in the table below.

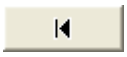
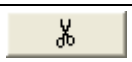
Button	Meaning
	To select the first line of a table use the button **First Line**.
	To select the previous line of a table use the button **Previous Line**.
	To select the next line of a table use the button **Next Line**.
	To select the last line of a table use the button **Last Line**.
	The button **Create a new Line** inserts new lines into the table.
	The button **Delete selected Line** deletes the selected line from the table.

Table 6: Table Line – Buttons

© Hilscher, 2009-2010

1.4.5 Status Bar

The **Status Bar** displays information about the current state of the DTM. The current activity, e.g. download, is signaled graphically via icons in the status bar.

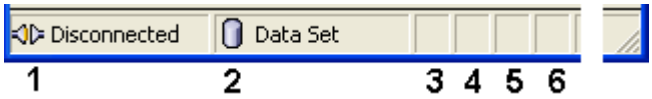
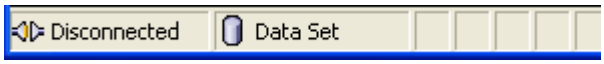


Figure 3: Status Bar – Status Fields 1 to 6

Status Field	Icon / Meaning
1	**DTM Connection States**
	Connected: Icon closed = Device is online
	Disconnected: Icon opened = Device is offline
2	**Data Source States**
	Data set: The displayed data are read out from the instance data set (database).
	Device: The displayed data are read out from the device.
3	**States of the instance Date Set**
	Valid Modified: Parameter is changed (not equal to data source).
4	**Changes directly made on the Device**
	Load/configure diagnosis parameters: Diagnosis is activated.
6	**Device Diagnosis Status**
	Save operation succeeded: The save operation has been successful.
	Further messages due to successful handling of device data.
	Firmware Download: Firmware Download is running
	Save operation failed: The save operation has failed.
	Further fail operation messages due to incorrect communication due to malfunction in the field device or its peripherals.

Table 7: Status Bar Icons [1]

Offline State



Save operation succeeded

Firmware Download

Firmware Download sussessful

Online State and Diagnosis

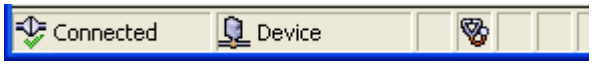


Figure 4: Status Bar Display Examples

© Hilscher, 2009-2010

2 Configuration of NT 100 as Gateway

The configuration of the NT 100 devices is explained exemplarily with the protocol conversion PROFINET IO Device to PROFIBUS-DP Master.

The device NT 100-RE-DP with one master license is necessary for the protocol conversion from PROFINET IO Device to PROFIBUS-DP Master.

The following steps have to be done to configure the device:

1. Start SYCON.net

➢ Select **Start > Programs > SYCON.net System Configurator > SYCON.net**

➧ SYCON.net is started

2. User Login

➢ In the window SYCON.net User Login press the button **OK** to login or enter your password and then press the button **OK** to login

➧ SYCON.net frame application appears

3. Create new project

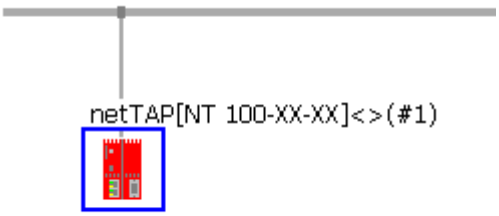
➢ Select the menu **File > New** to create a new project

➧ An empty project is created

4. Insert NT 100-XX-XX device into the configuration

➢ Go to the device catalog under vendor **Hilscher GmbH** to the category Gateway. Use drag and drop with the **NT 100-XX-XX** device to insert it at the bus line.

➧ The NT 100-XX-XX device appears in the project



© Hilscher, 2009-2010

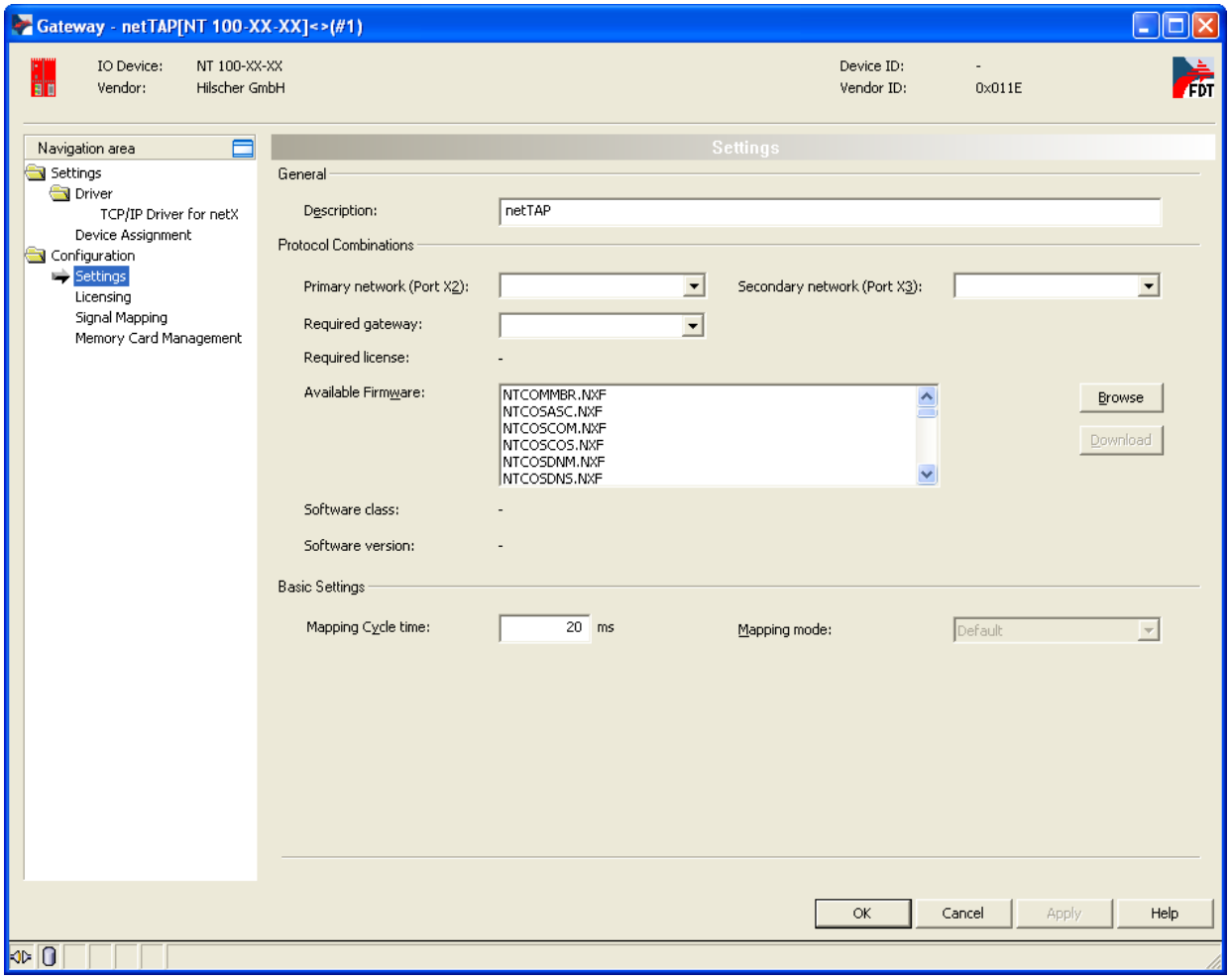
5. Open the Gateway configuration window

➢ Select from the context menu of the NT 100-XX-XX symbol (right mouse click) the entry **Configuration > Gateway**

➢ The Gateway configuration window opens

➢ Select in the Navigation area **Configuration** > **Settings**

➢ The configuration window **Settings** opens



© Hilscher, 2009-2010

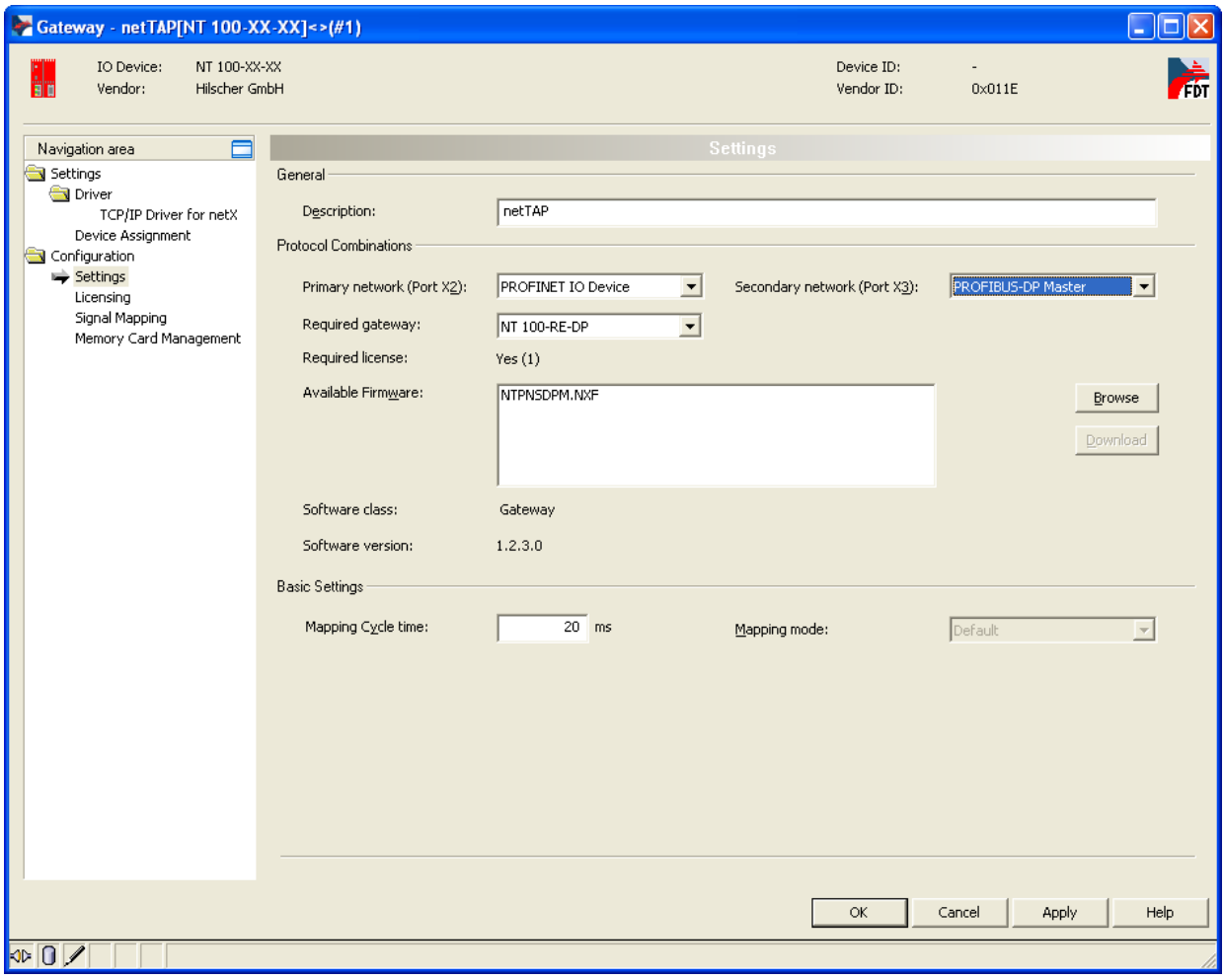
6. Select the protocol for the primary network and for the secondary network and apply

➢ Select in the window **Configuration > Settings** at **Protocol Combinations** for the **Primary Network (Port X2)** the protocol PROFINET IO Device

➢ Select then at **Protocol Combinations** for the **Secondary Network (Port X3)** the protocol PROFIBUS-DP Master

➢ Click the button **Apply**

➢ The Gateway configuration window shows the following



© Hilscher, 2009-2010

7. Close the Gateway configuration window

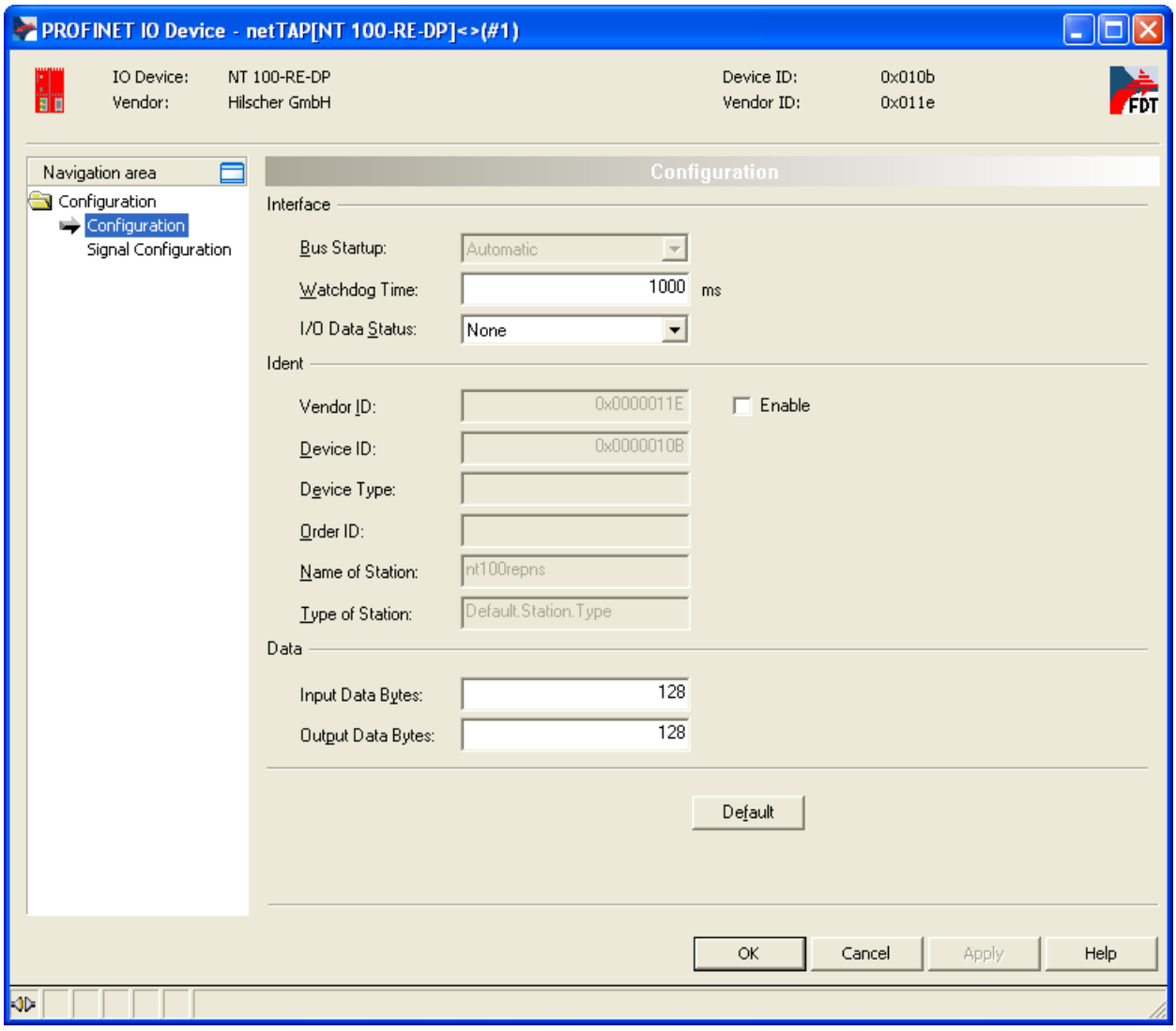
➢ Click on the button **OK**

↬ The Gateway configuration window closes

8. Configure the protocol at the primary network (Port X2)

➢ Select from the context menu of the NT 100-XX-XX symbol the entry **Configuration > PROFINET IO Device**

↬ The PROFINET IO Device configuration window opens



➢ Set the parameters. Set especially the number of Input Data Bytes and Output Data Bytes.

More information about PROFINET IO Device parameter is in section *Configuration PROFINET IO Device (Gateway)* on page 81.

© Hilscher, 2009-2010

9. Close the configuration window

➢ Click on the button **OK**

➧ The configuration window closes

10. Insert the PROFIBUS-DP Slave devices into the PROFIBUS network (Secondary network)

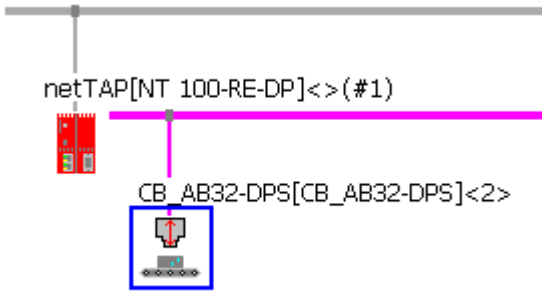
➢ Missing PROFIBUS-DP Slave devices can be added to the device catalog using the menu **Network > Import Device Descriptions**. Import the GSD file of the PROFIBUS-DP Slave.

➢ Then select **Network > Device Catalog**

➢ Then click the button **Reload Catalog**

➢ Go to the device catalog. Use drag and drop with one or more PROFIBUS-DP Slave devices to insert it/them at the PROFIBUS bus line.

➧ The PROFIBUS-DP Slave device icons appear at the PROFIBUS network line (Secondary network)



11. Configure the PROFIBUS-DP Slave device

➢ Open the configuration window with a double click on the device icon of the PROFIBUS-DP Slave

➧ The configuration window of the PROFIBUS-DP Slave device opens

More information about the configuration of PROFIBUS-DP Slave devices are in the document PROFIBUS_Generic_Slave_DTM_en.pdf in section *Configuration*

12. Open the PROFIBUS-DP Master (Port X3) configuration window

➢ Select from the context menu of the NT 100-XX-XX symbol the entry **Configuration > PROFIBUS-DP Master**

➧ The PROFIBUS-DP Master configuration window opens

© Hilscher, 2009-2010

13. Configure the PROFIBUS-DP Master

- Set the parameter. Set especially under **Configuration > Bus Parameter** the bus parameter and under **Configuration > Stationtable** the station addresses (stations addresses of the PROFIBUS-DP Slave devices)

  More information about the configuration of PROFIBUS-DP Master devices are in the document PROFIBUS_Master_netX_DTM_en.pdf in section *Configuration*

14. Close the configuration window

- Click on the button **OK**

⇨ The configuration window closes

15. Open the Gateway configuration window

- Select from the context menu of the NT 100-XX-XX symbol the entry **Configuration > Gateway**

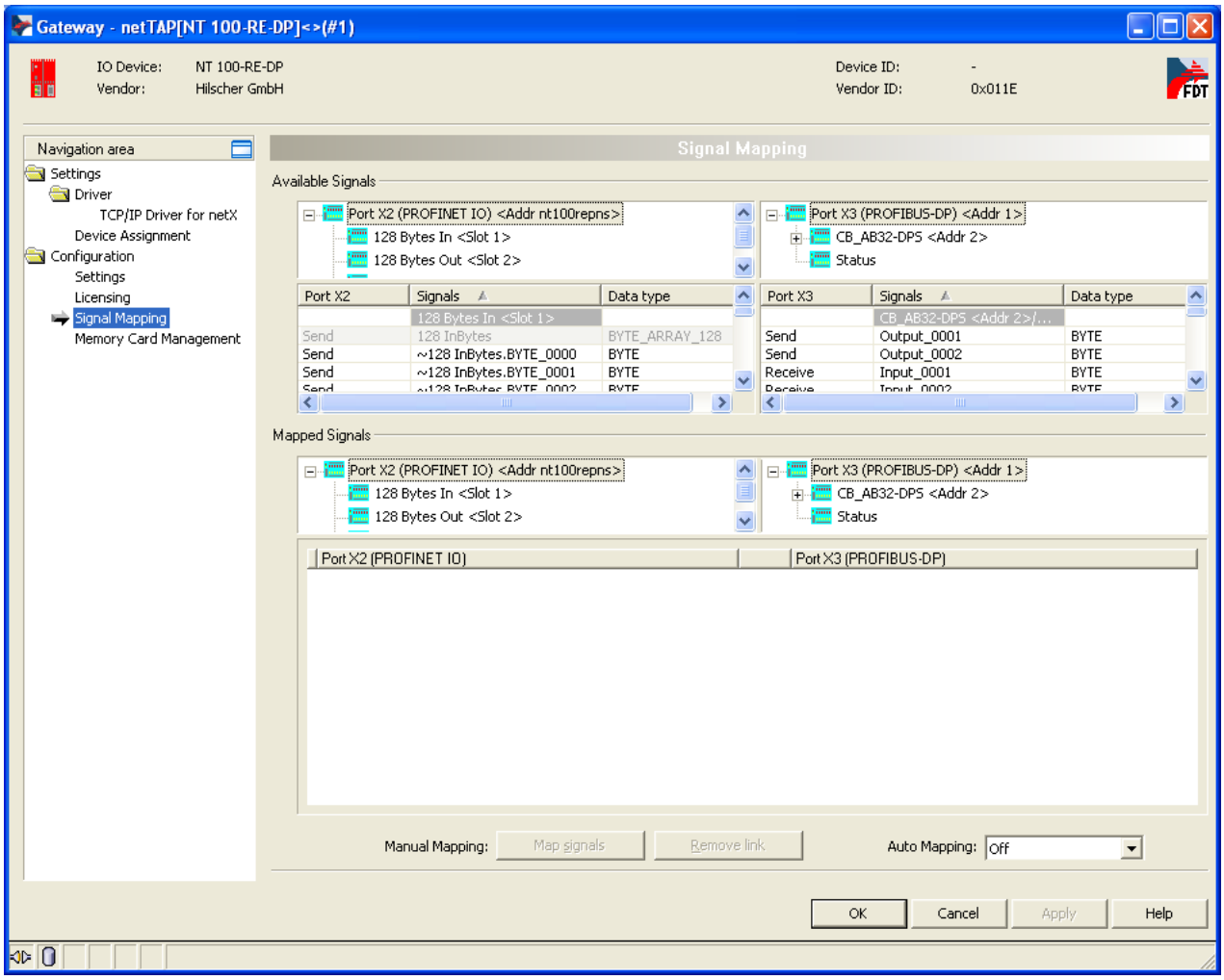
⇨ The Gateway configuration window opens

© Hilscher, 2009-2010

16. Open the Signal Mapping window

➢ Select **Configuration > Signal Mapping**

➢ The window Signal Mapping opens



© Hilscher, 2009-2010

17. Signal Mapping: Data transfer from Port X2 to Port X3

- Map the signals, which are received on Port X2 (Port X2 receive), with signals, which should be sent on Port X3 (Port X3 send).
- For this, mark the signal received (Port X2) and the signal to be sent (Port X3) and the click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the signal received (Port X2 receive) and drop it on the signal to be sent (Port X3 send)

18. Signal Mapping: Data transfer from Port X3 to Port X2

- Map the signals, which are received on Port X3 (Port X§ receive), with signals, which should be sent on Port X2 (Port X2 send).
- For this, mark the signal received (Port X§) and the signal to be sent (Port X2) and the click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the signal received (Port X3 receive) and drop it on the signal to be sent (Port X2 send)

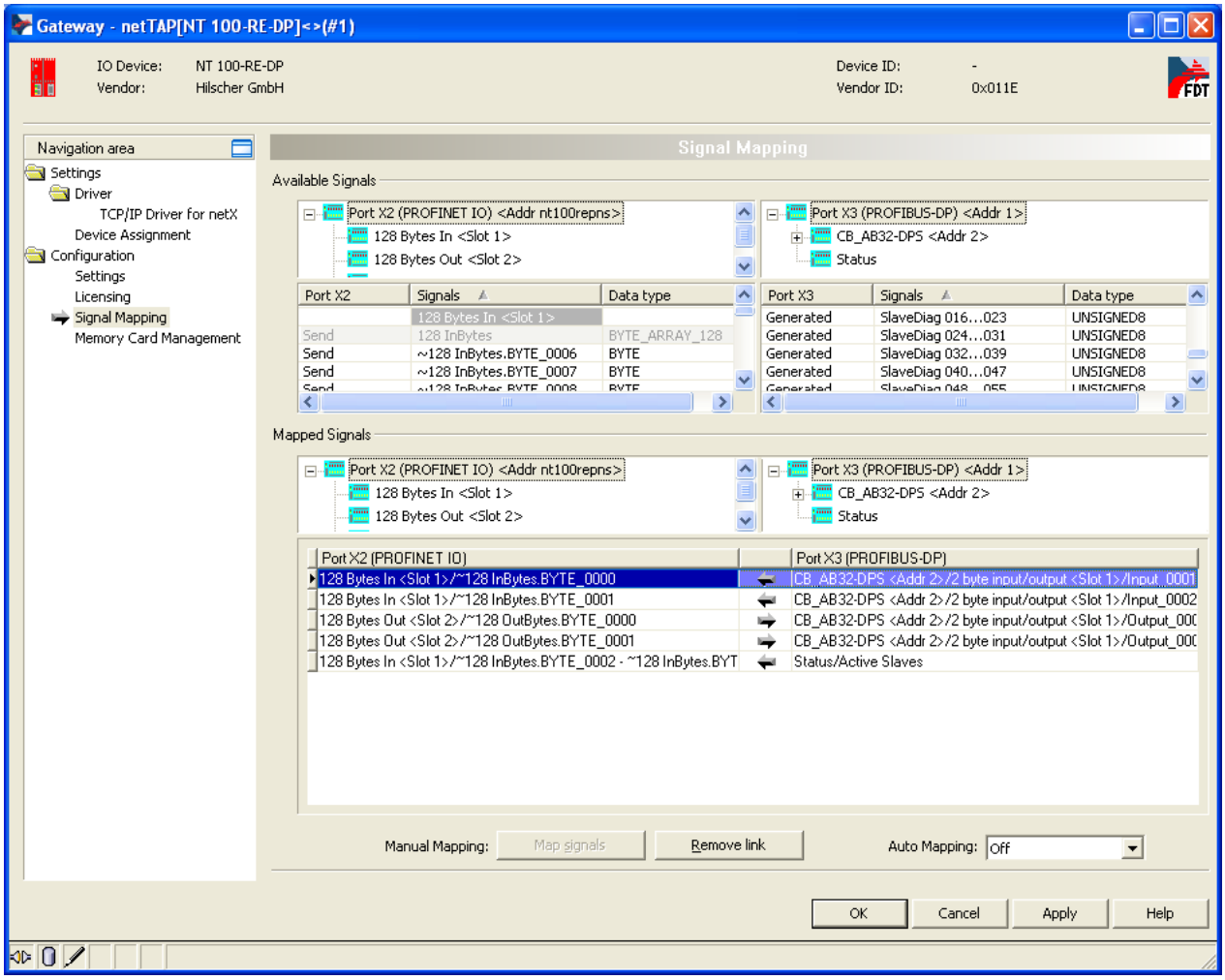
19. Signal Mapping: Transfer status info of Port X2 to Port X3

- If necessary, map the status information of Port X2 (Port X2 generated, which are generated device internal) to signals which should be sent on Port X3 (Port X3 send)
- For this mark the status signal (Port X2) and the signal which should be sent (Port X3) and click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the status signal (Port X2 generated) and drop it on the signal to be sent (Port X3 send)

© Hilscher, 2009-2010

20. Signal Mapping: Transfer status info of Port X3 to Port X2

- If necessary, map the status information of Port X3 (Port X3 generated, which are generated device internal) to signals which should be sent on Port X2 (Port X2 send)
- For this mark the status signal (Port X3) and the signal which should be sent (Port X2) and click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the status signal (Port X3 generated) and drop it on the signal to be sent (Port X2 send)
- An example of the I/O Data Mapping window after steps 17 to 20 shows the following figure



© Hilscher, 2009-2010

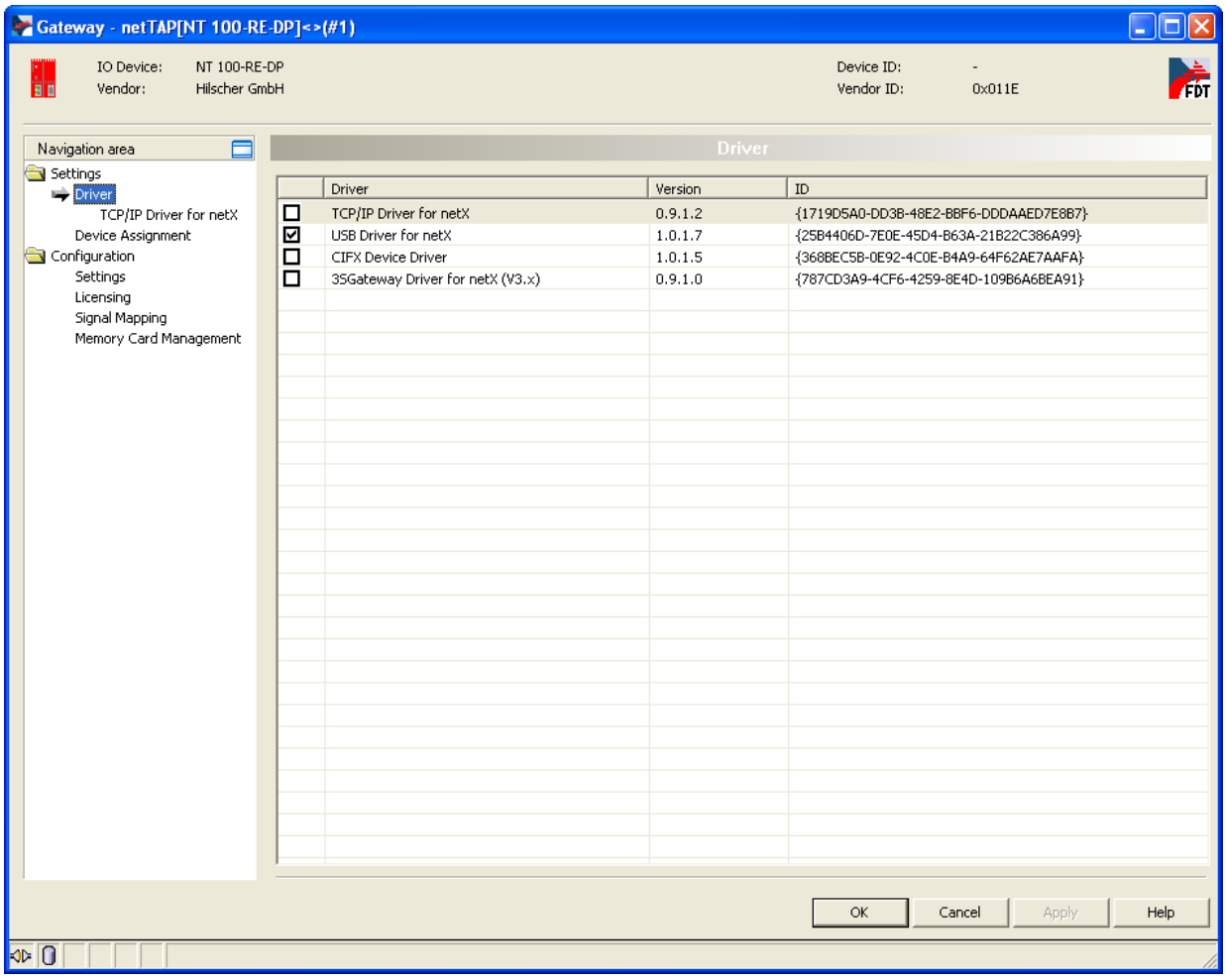
21. Connect the USB cable

➢ Connect with an USB cable the USB connector of the netTAP device with the USB connector of the PC.

22. Select the USB driver

➢ Select **Settings > Driver** and then check **USB Driver for netX**

➢ The following figure shows the selected USB Driver

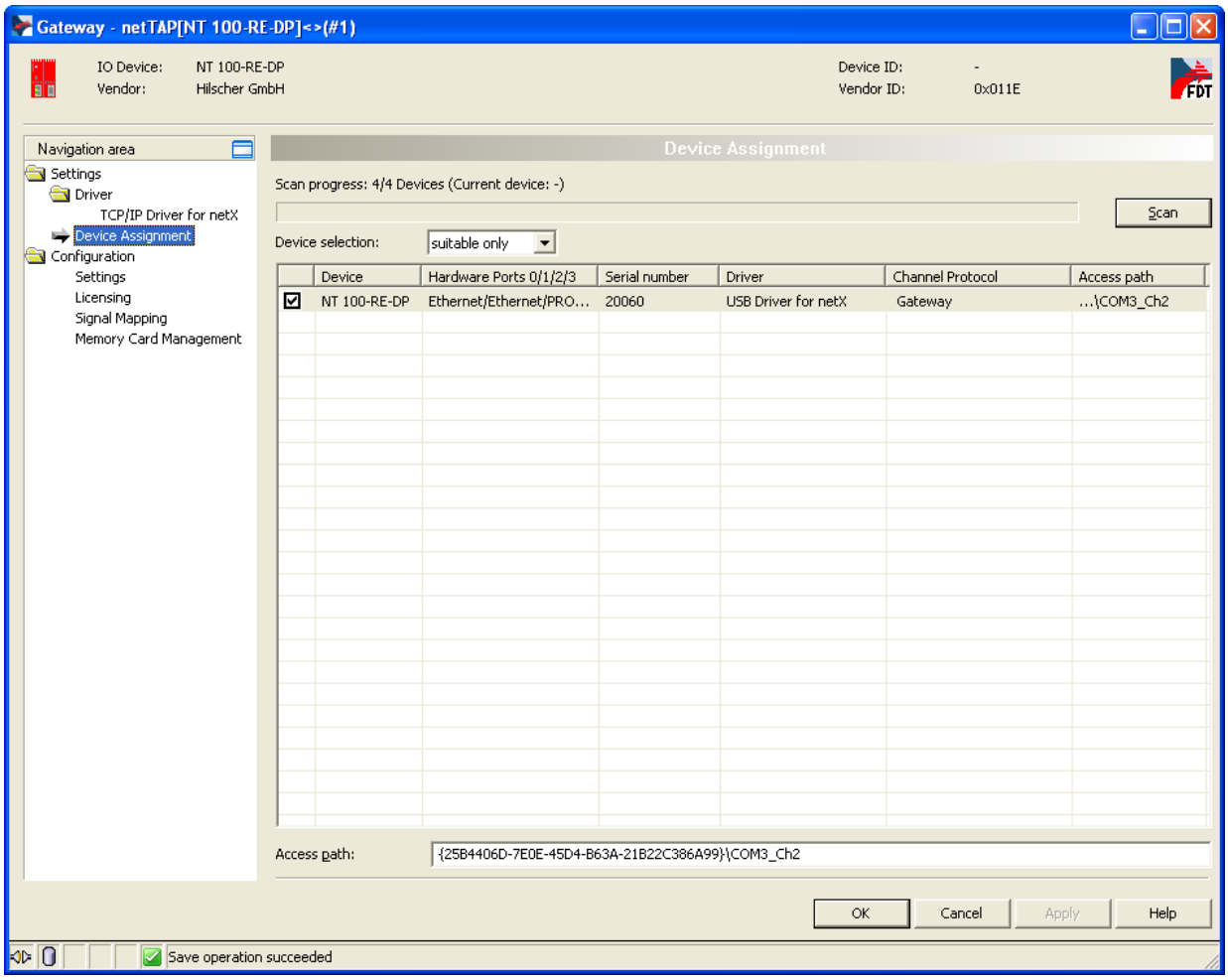


© Hilscher, 2009-2010

23. Device Assignment

- Select **Settings > Device Assignment**
- Click the button **Scan**
- Check the found netTAP device

The following figure shows the assigned device



© Hilscher, 2009-2010

24. Load Firmware

- Select **Configuration > Settings**
- Mark the Firmware file in section Firmware Download
- Click the button **Download**

↬ The firmware is loaded into the netTAP device

25. Close the Gateway configuration window

- Click the button **OK**

↬ The Gateway configuration window closes

26. Download configuration

- Select from the context menu of the NT 100-XX-XX symbol the entry **Download**
- Answer the security question with Yes, if the download should start

↬ The configuration is transferred info the netTAP device

27. Save project

- Select **File > Save**, enter a file name and save the project

↬ the project file is stored

28. Device Description File for the configuration of the Master

- Use for the configuration of the PROFINET IO Controller the device description file GSDML-V2.1-HILSCHER-NT 100-RE PNS-20090123.xml from the CD from the directory EDS/PROFINET.

© Hilscher, 2009-2010

3 Settings

3.1 Overview Settings

Settings Dialog Panes

The table below gives an overview for the individual **Settings** dialog panes descriptions:

Section	Subsection	Page
Driver		30
	Selecting the Driver	30
Device Assignment		31
	Scanning for Devices	32
	Selecting the Device	34

Table 8: Descriptions Settings

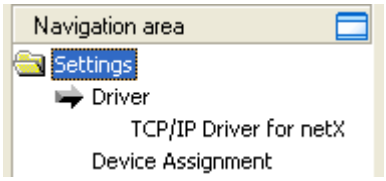


Figure 5: Navigation Area - Settings

Note: To edit the **Settings** dialog panes you need *User Rights* for “Maintenance”.

© Hilscher, 2009-2010

3.2 Driver

The **Driver** dialog pane displays the driver/s to be used for a netGateway DTM to device communication connection.

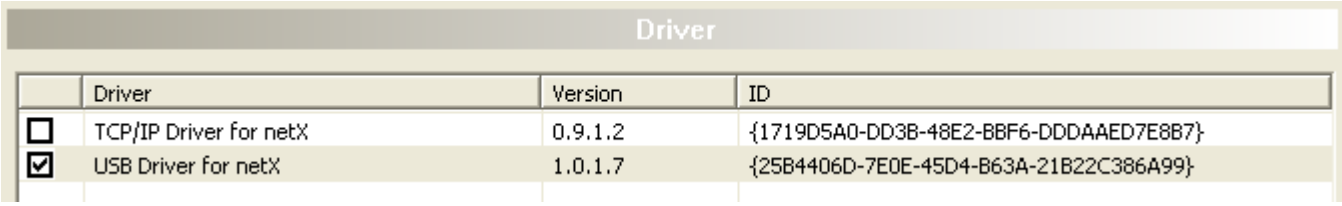


Figure 6: Driver Selection List

Parameter	Meaning
Driver	Name of the driver
Version	Version of the driver
ID	ID of the driver (driver identification)

Table 9: Driver Selection List Parameters

3.2.1 Selecting the Driver

To establish a connection you need to select a driver first.

Note: Not all the drivers displayed necessarily support the device. The used driver must be supported by the device or must be available for the device.

To select the driver/s to be used:

1. Select **Settings > Driver** in the navigation area.

   ↬ The available drivers are listed on the **Driver** dialog pane.

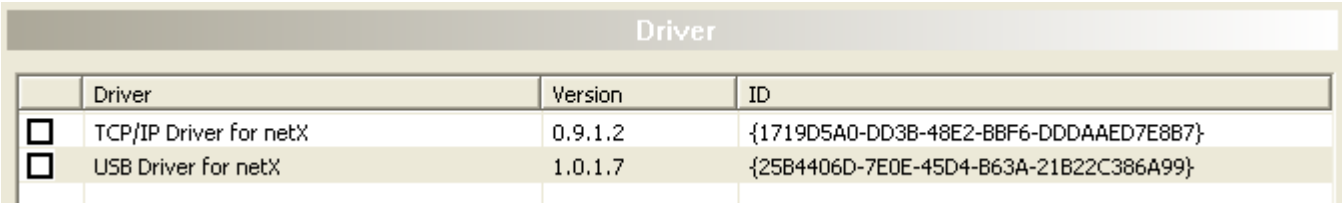


Figure 7: Driver Selection List - recognized or installed Drivers

2. Check the checkbox for the driver/s in the selection list.

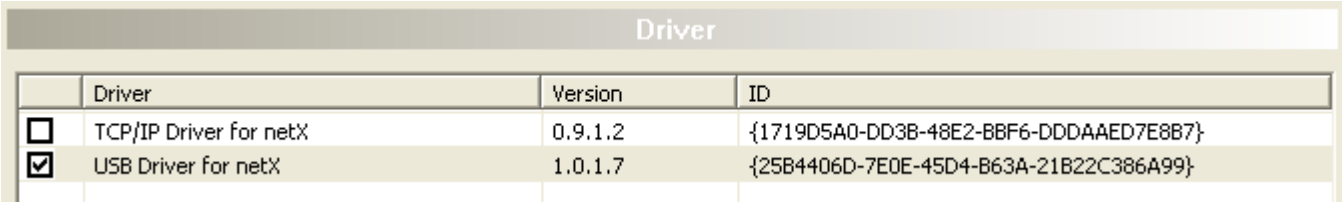


Figure 8: Driver Selection List - Driver/s selected

© Hilscher, 2009-2010

3.3 Device Assignment

Note: To establish a connection from the netGateway DTM to a netTAP NT 100 device, you first must assign a netTAP NT 100 device to the netGateway DTM. Therefore you must assign the netTAP NT 100 device to the selected driver.

Therefore in the **Device Assignment** dialog pane you must scan for the netTAP NT 100 device and select it.

If the device shall get a new firmware:

1. first you must scan for the device and select it,
2. then you must download a firmware to the device

© Hilscher, 2009-2010

3.3.1 Scanning for Devices

1. Select **Settings > Device Assignment** in the navigation area.

↬ The dialog pane **Device Assignment** is displayed.

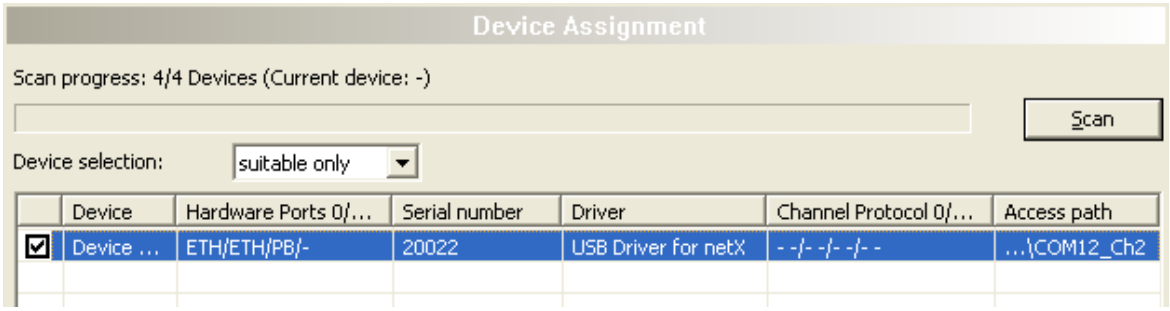


Figure 9: Device Assignment - detected Devices (The name of the device class is displayed.) – Example for a device with firmware*

2. Select the list box entry **suitable only**.
3. Select the **Scan** button, to start the scanning process.

↬ Under **Device Selection** all devices are displayed, which can be connected to the netGateway DTM via the preselected driver.

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value / Value
Device selection	Selecting **suitable only** or **all** devices.	suitable only, all
Device	Device name.	
Hardware Port 0/1/2/3	Shows, which hardware is assigned to which communication interface.	
Serial number	Serial number of the device	
Driver	Name of the driver	
Channel Protocol	Shows, which firmware is loaded to which device channel. The data for the used channel consists of the protocol class and the communication class. a.) For devices without firmware: Undefined Undefined, b.) For devices with firmware: e. g. PROFIBUS Master	
Access path (under Device selection last column on the right)	Depending on the used driver in the column **Access path** different data to the device are displayed.	Depending on the device and on the driver: board or channel number, IP address or COM interface
Access path (at the lower side of the dialog pane)	If under **Device selection** the check box for a device is checked, under **Access path** (at the lower side of the dialog pane) the driver identification or depending on the used driver additional data to the device.	driver identification (ID) depending on the device and on the driver: board or channel number, IP address or COM interface

Table 10: Parameters of the Device Assignment

© Hilscher, 2009-2010

3.3.2 Selecting the Device

Note: A connection with the netGateway DTM can only be established with one netTAP NT 100 device.

To select the physical netTAP NT 100 device:

1. In the **Device Assignment** pane in the selection list **Device selection** check the check box for the device.

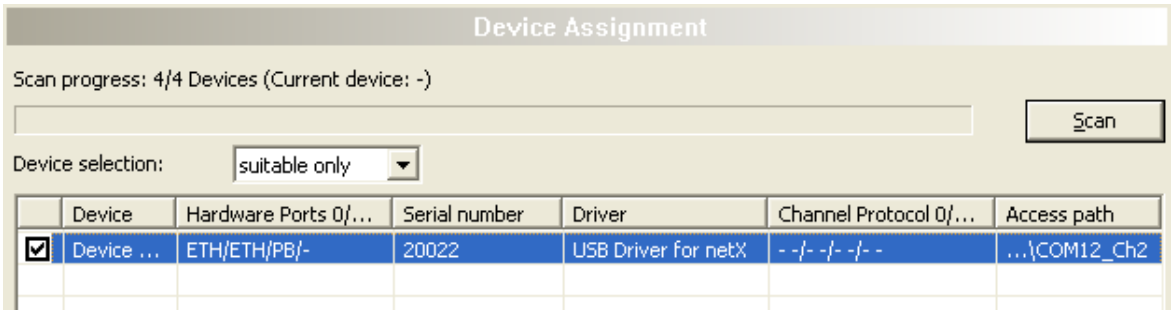


Figure 10: Device Assignment - Selecting the Device

- Under **Access path** (below in the dialog pane) the access path to the device, e. g. the driver identification, or depending on the used driver additional access data of the device are displayed.

2. Select the **Apply** button, to apply the selection.

© Hilscher, 2009-2010

4 Configuration

4.1 Overview Configuration

Configuration Dialog Panes

The table below gives an overview for the Configuration dialog panes descriptions:

Section	Page
Settings	*36*
Licensing	*37*
Signal Mapping	*45*
Memory Card Management	*52*

Table 11: Descriptions of the Dialog Panes Configuration

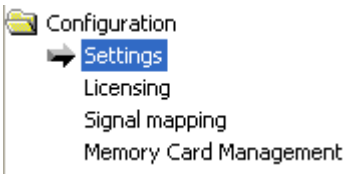


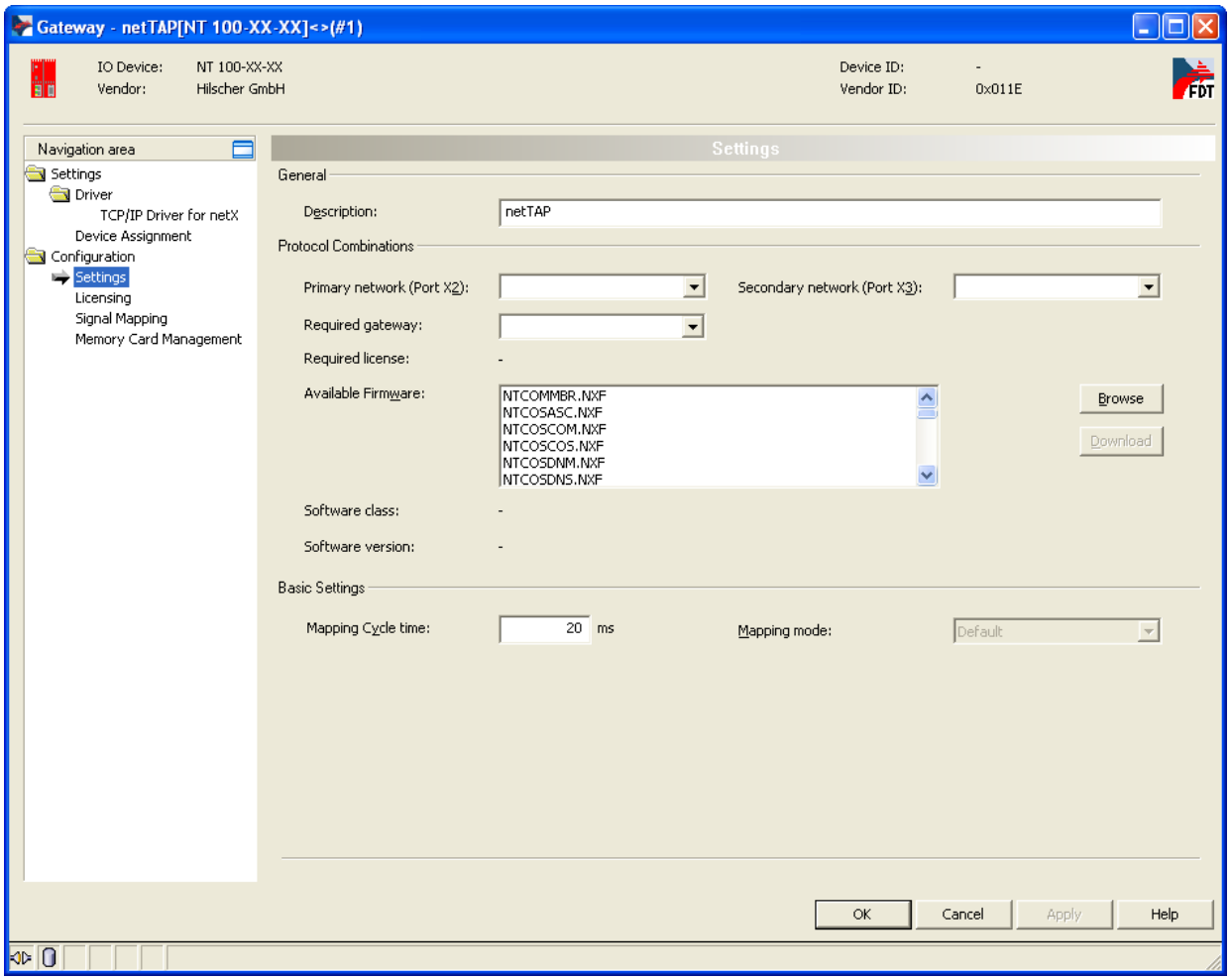
Figure 11: Navigation Area - Configuration

© Hilscher, 2009-2010

4.2 Settings

In the **Settings** pane you can:

- select the protocol for Port X2 respectively for Port X3
- transfer the firmware into the netTAP NT 100 device
- set the cycle time of the Gateway



© Hilscher, 2009-2010

4.3 Licensing

In the **Licensing** pane you can:

- check, which licenses are present in the device,
- order licenses,
- transfer license files to the device.

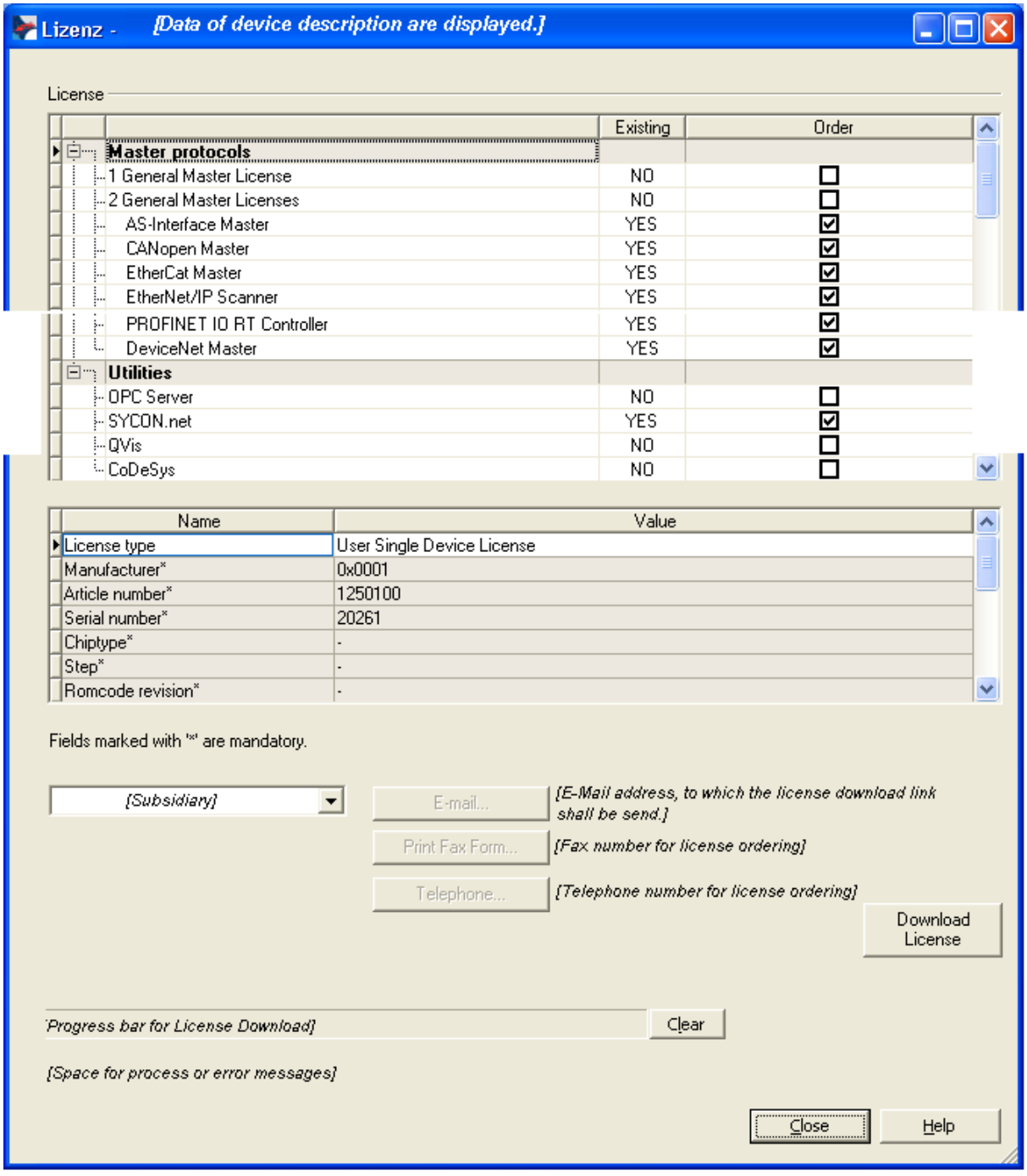


Figure 12: License Pane

© Hilscher, 2009-2010

4.3.1 Which licenses are present in the Device?

Check in the **License** pane which licenses are present in the device.

- Therefore click to the **Master Protocols** or **Utilities** folder to expand it.
- The column **Existing** indicates which licenses are present in the device.

License for Protocols

1 General Master License.

On the device maximally 1 communication protocol with master function can be implemented.

2 General Master Licenses.

On the device maximally 2 communication protocols with master function can be implemented.

License for Utilities

SYCON.net, OPC Server, QVis or CoDeSys

© Hilscher, 2009-2010

4.3.2 How to order a license

To order a license proceed as follows:

- Assign the device to the DTM,
- Select the required licenses,
- Enter the ordering data and
- Send your order via E Mail, Fax or Telephone.

4.3.2.1 Assigning the Device to the DTM

Note: You first need to assign the device to the DTM. Only after this the device data and the available licenses are displayed in the **License** dialog.

1. Open the DTM configuration dialog via double click to the device symbol.
2. Select one or several driver/s.

➢ Select **Settings > Driver** in the navigation area.

➢ In the **Driver** pane check the checkbox for the driver/s.

3. Configure the driver, if necessary.

➢ Select in the navigation area **Settings > Driver >** [*Name of the assigned driver*].

➢ Select **Settings > Driver** > [*Name of the assigned driver*] in the navigation area.

➢ In the drive dialog pane configure the driver settings.

4. Scan for the device/s.

➢ Select **Settings > Device Assignment** in the navigation area.

➢ In the **Device Assignment** pane select the list box entry **suitable only** or **all**.

➢ Select the **Scan** button, to start the scanning process.

5. Select the device and apply the selection.

➢ In the **Device Assignment** pane in the selection list **Device selection** check the check box for the device.

➢ Select the **Apply** button, to apply the selection.

6. Close the DTM configuration dialog via **OK**.

For details to the device assignment, refer to section *Overview Settings* on page 29.

© Hilscher, 2009-2010

4.3.2.2 Selecting the License

1. Open the **License** pane:

➢ Therefore open the context menu, via right mouse click to the device symbol.

➢ From the context menu select **Additional Functions > License**.

In the **License** pane:

2. Under **License** click to the **Master Protocols** folder to expand it.
3. In the column **Order** check the checkbox for the desired number of licenses:
   1 General Master License or *2 General Master Licenses*.

And / Or:

4. Under **License** click to the **Utilities** folder to expand it.
5. In the column **Order** check the checkbox for:
   S*YCON.net*, *OPC Server*, *QVis* or *CoDeSys*.

© Hilscher, 2009-2010

4.3.2.3 Entering the Order Data

1. In the **License** pane under **Name** enter any obligatory order data.
2. Select in the list field **[*subsidiary*]** the entry for the subsidiary to which the order shall be send.

Device Data (order data read from the device)

The following order data read from the device are displayed in the **License** pane:

- Manufacturer
- Device number
- Serial number
- Chiptype
- Step (chip revision)
- Romcode revision
- Checksum (checksum of the device data)

Data to manage the Order

Mandatory data to the order request (editable fields in the **License** pane):

- Licensetype (User Single Device License):
- First Name
- Surname
- E Mail (E-Mail address, to which the license download link shall be send.)
- Telephone
- Fax
- Company
- Addresse
- Country
- City, State, Zip

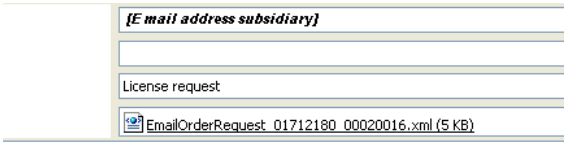
Additional order data, not mandatory (editable fields in the **License** pane):

- Order number
- Value added tax identification number

© Hilscher, 2009-2010

4.3.2.4 Ordering the License by E Mail

- Click to the button **E Mail**.
- A prepared ordering E Mail **License request** is opened, to the E Mail address of the subsidiary, to which the order shall be send.
- This ordering E Mail contains the automatically generated file *EmailOrderRequest_[Devicenumber]_[Serialnumber].xml* with a summary info of the order information.




```
==================================================
                netX License Order
==================================================
{Address data subsidiary}
Licensee Information
--------------------------------------------------
First Name:         Hans
Surname:            Mustermann
e-Mail:             mustermann@musterfirma.com
Telephone:          004971156678
Fax:                0049711566790
Company:            Musterfirma
Address:            Musterstrasse 7
Country:            Deutschland
City Zip:           67000
Order Number:
Tax Ident. Number:

License Type
--------------------------------------------------
User Single Device License

Device Information
--------------------------------------------------
Manufacturer:  00000001
Device Number: 01712180
Serial Number: 00020016
Chip Type:     00000002
Step:          00000000
Romcode Revision: 00000002

Ordered Licenses
--------------------------------------------------
Master Protocols:

>    1 General Master License
>    AS-Interface Master
>    CANopen Master
>    EtherCat Master
>    EtherNet/IP Scanner
>    PROFIBUS Master
>    PROFINET IO RT Controller
>    DeviceNet Master

Utilities:

>    SYCON.net
```

Figure 13: Example: ordering E Mail License request

- Send the ordering E Mail **License request**.
- The order process is complete.

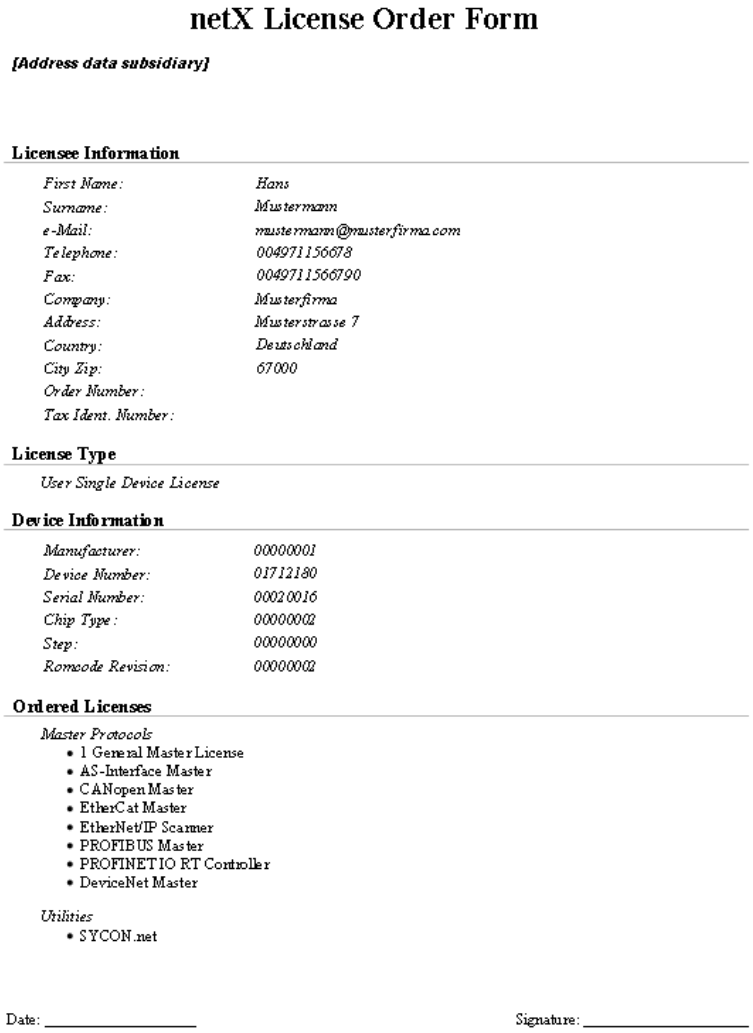
© Hilscher, 2009-2010

4.3.2.5 Ordering the License by Fax or by Telephone:

➢ Click to the button **Print Fax Form** or **Telephone**.

Note: If your browser does not display the order data or the window **Move Element** or **Copy Element** are displayed, check the safety settings of your system.

➧ The summary of the ordering data *PrintOrderRequest_ [Devicenumber]_[Serialnumber].html* **netX License Order Form** is opend in a browser window.



netX License Order Form

[Address data subsidiary]

Licensee Information

First Name:	*Hans*
Surname:	*Mustermann*
e-Mail:	*mustermann@musterfirma.com*
Telephone:	*0049711566678*
Fax:	*0049711566790*
Company:	*Musterfirma*
Address:	*Musterstrasse 7*
Country:	*Deutschland*
City Zip:	*67000*
Order Number:	
Tax Ident. Number:	

License Type

User Single Device License

Device Information

Manufacturer:	*00000001*
Device Number:	*01712180*
Serial Number:	*00020016*
Chip Type:	*00000002*
Step:	*00000000*
Romcode Revision:	*00000002*

Ordered Licenses

Master Protocols

- 1 General Master License
- AS-Interface Master
- CANopen Master
- EtherCat Master
- EtherNet/IP Scanner
- PROFIBUS Master
- PROFINET IO RT Controller
- DeviceNet Master

Utilities

- SYCON.net

Date: _____ Signature: _____

Figure 14: Example: order data form netX License Order Form

➢ Print out the order data form **netX License Order Form** and send it by fax.

Or:

➢ Keep ready the order data form and communicate the order data via telephone.

➧ The order process is complete.

© Hilscher, 2009-2010

4.3.3 How to get the License and transfer it to the Device

Licenses are delivered as license files.

Note: License files can only be delivered via E Mail.

➧ The license file is delivered via E Mail. The E Mail contains a link to download the license file.

After the license file has been delivered you must transfer the received license file into the device.

1. Download the license file **.nxl* to your system.

➢ Therefore open the E Mail.

➢ Click to the **Link for the Download of the license file** and download it to your system.

2. Transfer the license file **.nxl* to the device.

➢ In the **License** pane click to the button **Download License**.

➢ In the file selection window **Open** select the license file *netX License Files (*.nxl)* and click to the button **Open**.

➧ Then the license is present in the device an will be activated at the next reset of the device.

© Hilscher, 2009-2010

4.4 Signal Mapping

The signal mapping determines which user data of the primary network are assigned to which user data of the secondary network and vice versa.

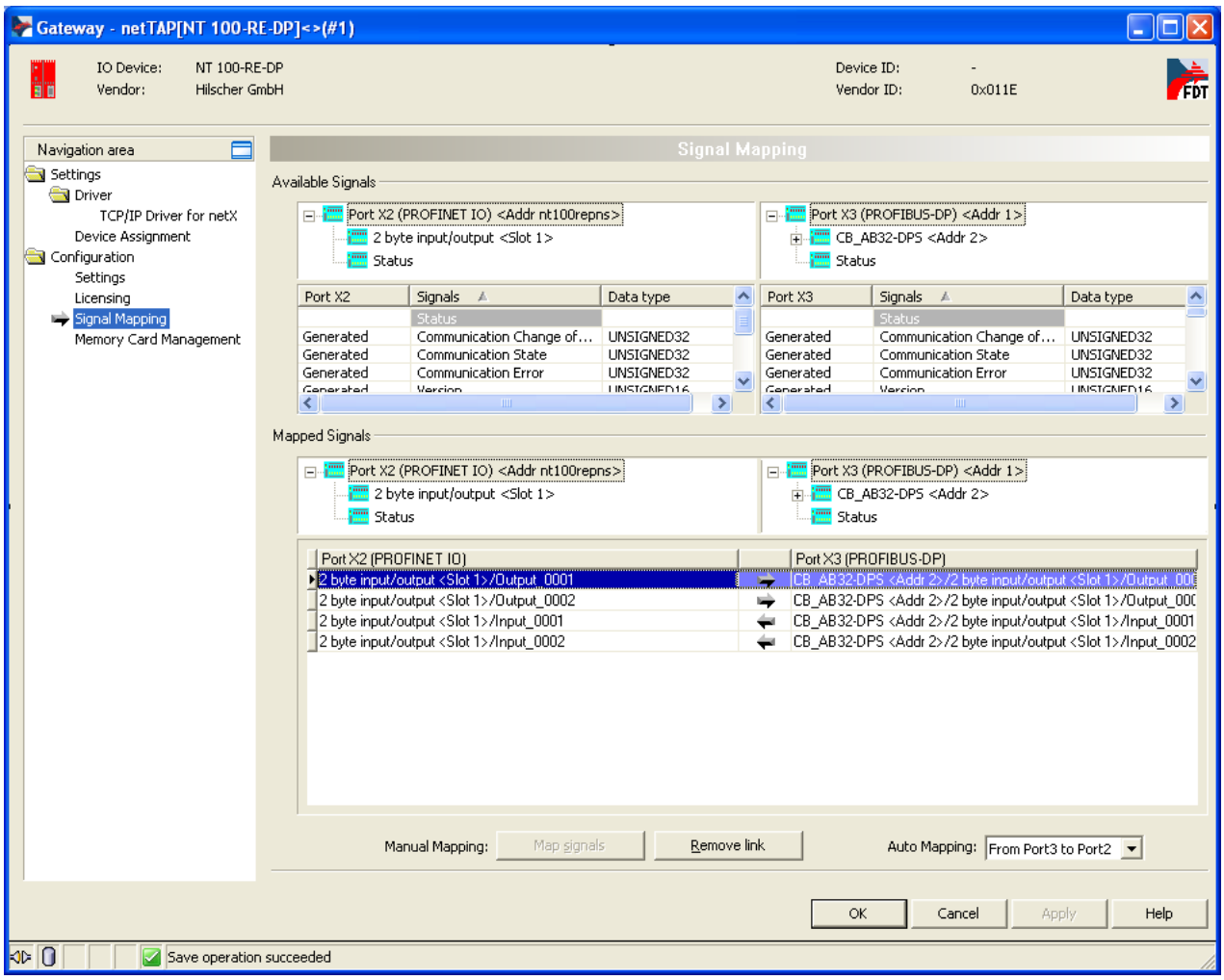
With the mapping

- the received data from the primary network are mapped to the send data of the secondary network,
- the received data from the secondary network are mapped to the send data of the primary network,
- status information of the primary network are mapped to the send data of the secondary network,
- status information of the secondary network are mapped to the send data of the primary network.

With this mapping a device internal list in generated. The device work each x ms through this list and copies the data.

© Hilscher, 2009-2010

On the pane I/O Data Mapping you can map the I/O Data of port X2 to the I/O data of port X3.



The I/O mapping is done the following way:

Data transfer from Port X2 to Port X3

- Map the signals, which are received on Port X2 (Port X2 receive), with signals, which should be send on Port X3 (Port X3 send).
- For this, mark the signal received (Port X2) and the signal to be send (Port X3) and the click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the signal received (Port X2 receive) and drop it on the signal to be send (Port X3 send)

© Hilscher, 2009-2010

Data transfer from Port X3 to Port X2

- Map the signals, which are received on Port X3 (Port X§ receive), with signals, which should be send on Port X2 (Port X2 send).
- For this, mark the signal received (Port X§) and the signal to be send (Port X2) and the click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the signal received (Port X3 receive) and drop it on the signal to be send (Port X2 send)

Transfer status info of Port X2 to Port X3

- If necessary, map the status information of Port X2 (Port X2 generated, which are generated device internal) to signals which should be send on Port X3 (Port X3 send)
- For this mark the status signal (Port X2) and the signal which should be send (Port X3) and click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the status signal (Port X2 generated) and drop it on the signal to be send (Port X3 send)

Transfer status info of Port X3 to Port X2

- If necessary, map the status information of Port X3 (Port X3 generated, which are generated device internal) to signals which should be send on Port X2 (Port X2 send)
- For this mark the status signal (Port X3) and the signal which should be send (Port X2) and click the button **Map Signals**
  Or
  Map the signal using drag and drop. For this, drag the status signal (Port X3 generated) and drop it on the signal to be send (Port X2 send)

Auto mapping

The signal mapping can also be done automatically be the configuration software. Set **Auto Mapping** to **From Port X3 to Port X2** and then click on **Apply**. The automatic signal mapping is done.

© Hilscher, 2009-2010

4.4.1 Status Information

The status of the remote network coupled by the netTAP NT 100 can be mapped into the I/O data. It contains the items shown in the figures for master and slave and are described below.

Signals	Data type	Port X3
Status		
Active Slaves	UNSIGNED32	Generated
Communication Change of State	UNSIGNED32	Generated
Communication Error	UNSIGNED32	Generated
Communication State	UNSIGNED32	Generated
Configured Slaves	UNSIGNED32	Generated
Error Count	UNSIGNED32	Generated
Error Log Indicator	UNSIGNED32	Generated
Faulted Slaves	UNSIGNED32	Generated
Host Watchdog	UNSIGNED32	Generated
Slave Error Log Indicator	UNSIGNED32	Generated
Slave State	UNSIGNED32	Generated
Version	UNSIGNED16	Generated
Watchdog Timeout	UNSIGNED16	Generated

Figure 15: Status information of the Master

Signals	Data type	Port X2
Status		
Communication Change of State	UNSIGNED32	Generated
Communication Error	UNSIGNED32	Generated
Communication State	UNSIGNED32	Generated
Error Count	UNSIGNED32	Generated
Error Log Indicator	UNSIGNED32	Generated
Host Watchdog	UNSIGNED32	Generated
Version	UNSIGNED16	Generated
Watchdog Timeout	UNSIGNED16	Generated

Figure 16: Status information of the Slave

Some status information belong to the master firmware and to the slave firmware status and are marked with “Master and Slave”), others belong only to the master firmware status and are marked with “Master”.

Number of Active Slaves (Master)

The firmware maintains a list of slaves within the remote network to which the remote network’s master has successfully opened a connection. Ideally, the number of active slaves is equal to the number of configured slaves.

This field holds the number of active slaves.

© Hilscher, 2009-2010

Communication Change of State (Master and Slave)

The communication change of state register contains information about the current operating status of the communication channel and its firmware.

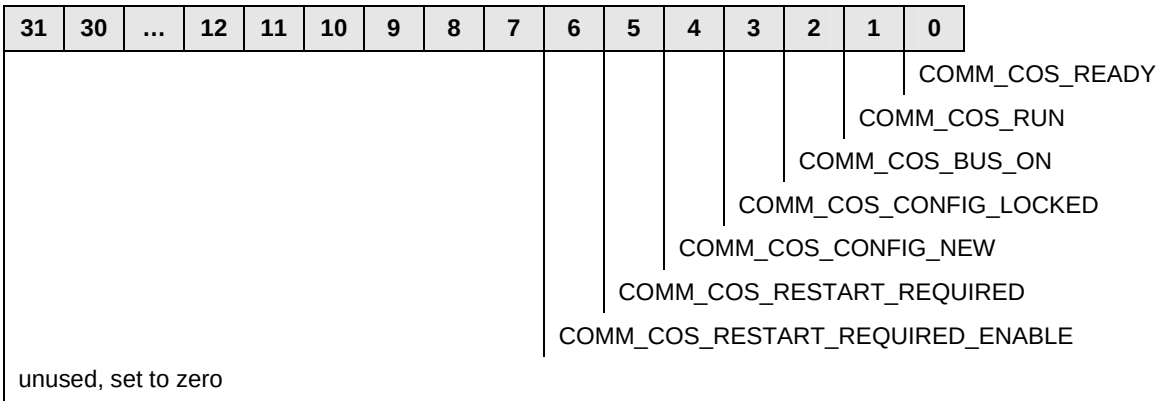


Table 12: Communication Change of State

The Communication Change of State Flags have the following meaning:

Status	Meaning
Ready (Bit 0)	The Ready flag is set as soon as the protocol stack is started properly. Then the protocol stack is awaiting a configuration. As soon as the protocol stack is configured properly, the Running flag is set, too.
Running (Bit 1)	The Running flag is set when the protocol stack has been configured properly. Then the protocol stack is awaiting a network connection. Now both the Ready flag and the Running flag are set.
Bus On (Bit 2)	The Bus On flag is set to indicate to the host system whether or not the protocol stack has the permission to open network connections. If set, the protocol stack has the permission to communicate on the network; if cleared, the permission was denied and the protocol stack will not open network connections.
Configuration Locked (Bit 3)	The Configuration Locked flag is set, if the communication channel firmware has locked the configuration database against being overwritten. Re-initializing the channel is not allowed in this state.
Configuration New (Bit 4)	The Configuration New flag is set by the protocol stack to indicate that a new configuration became available, which has not been activated. This flag may be set together with the Restart Required flag.
Restart Required (Bit 5)	The Restart Required flag is set when the channel firmware requests to be restarted. This flag is used together with the Restart Required Enable flag below. Restarting the channel firmware may become necessary, if a new configuration was downloaded from the host application or if a configuration upload via the network took place.
Restart Required Enable (Bit 6)	The Restart Required Enable flag is used together with the Restart Required flag above. If set, this flag enables the execution of the Restart Required command in the netX firmware.

© Hilscher, 2009-2010

Communication State (Master and Slave)

The communication state field contains information regarding the current network status of the communication channel of the remote network. Depending on the implementation of the protocol stack of the remote network, all or a subset of the definitions below is supported.

Status	Value
Unknown	0
Offline	1
Stop	2
Idle	3
Operate	4

Communication Channel Error (Master and Slave)

This field holds the current error code of the communication channel of the remote network. If the cause of error is resolved, the communication error field is set to zero again.

All values different from 0 indicate that an error has occurred.

Errors may be signaled either from the operating system rcX or from the used protocol.

Number of Configured Slaves (Master)

The firmware maintains a list of slaves within the remote network to which the master has to open a connection. This list is derived from the configuration database created by SYCON.net. This field holds the number of configured slaves.

Error Count (Master and Slave)

This field holds the total number of errors detected since power-up within the remote network, respectively after reset. The protocol stack counts all sorts of errors in this field no matter if they were network related or caused internally. After power cycling, reset or channel initialization this counter is being cleared again.

Error Log Indicator (Master and Slave)

Note: This field is not yet supported.

© Hilscher, 2009-2010

Number of Faulted Slaves (Master)

If a slave of the remote network encounters a problem, it can provide an indication of the new situation to the master in certain Fieldbus systems. As long as those indications are pending and not serviced, this field holds a value unequal to zero. If no more diagnostic information is pending, the field is set to zero.

Host Watchdog (Master and Slave)

Note: This field is not yet supported.

Slave Error Log Indicator (Master)

Note: This field is not yet supported.

Slave State (Master)

The slave state field indicates whether the master of the remote network is in cyclic data exchange to all configured slaves. In case there is at least one slave missing or if the slave has a diagnostic request pending, the status is set to FAILED. For protocols that support non-cyclic communication only, the slave state is set to OK as soon as a valid configuration is found.

Status	Value
Undefined	0
Ok	1
Failed	2

Version (Master and Slave)

The version field holds the version number of this structure. The value is 1.

Watchdog Timeout (Master and Slave)

This field holds the configured watchdog timeout value of the protocol stack of the remote network. The value is specified in milliseconds.

© Hilscher, 2009-2010

4.5 Memory Card Management

Note: This function is only available for the netTAP NT 100 device family, because these devices have a MMC card slot.

In the **Memory Card management** pane you can:

- backup the firmware and the configuration from the netTAP device to an MMC memory card
- restore the firmware and the configuration from the MMC memory card into the netTAP device

On the pane MMC Management you can backup and restore the firmware and the configuration to respectively from an MMC card.

The **Folder** on the pane shows the directory structure of the file system of the netTAP device named **SYSVOLUME** and if an MMC card is inserted also the directory structure of the MMC card named **SDMMC**.

Backup: To backup the firmware and configuration files insert a formatted (Format FAT) MMC card into the MMC slot of the netTAP NT 100 device. When the directory named **SDMMC** is shown, then click on the button **Backup** to copy the firmware and configuration files from the netTAP NT 100 device to the MMC card.

Restore: To load the firmware and configuration files from an MMC card into the netTAP NT 100 device, insert an MMC card into the MMC slot of the netTAP NT 100 device which contains the firmware and configuration files. When the directory named SDMMC is shown, then click the button Restore to copy the firmware and configuration files from the MMC card into the netTAP NT 100 device.

© Hilscher, 2009-2010

4.6 Configuration EtherCAT Master

The netTAP NT 100 device respectively the netBRICK NB 100 device as EtherCAT Master needs a configuration, e. g. information about how many EtherCAT Slave devices with how many input and output data are to be connected.

1. Insert EtherCAT Slave devices

- Drag from the device catalog EtherCAT Slave device(s) and drop it to the bus line of the EtherCAT Master.

2. Configure EtherCAT Slave devices

- Open the configuration dialog for each EtherCAT Slave device and configure the device.

  Information about how to configure the EtherCAT Slave device is in the operating instruction manual „Generic DTM for EtherCAT Slave devices“ and has the file name EtherCAT_GenericSlave_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure EtherCAT Master

- Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > EtherCAT Master**

  Information about how to configure the EtherCAT Master device is in the operating instruction manual „DTM for EtherCAT Master devices“ and has the file name EtherCAT_Master_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.7 Configuration EtherCAT Slave

The netTAP NT 100 device respectively the netBRICK NB 100 device as EtherCAT Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > EtherCAT Slave**

➧ The EtherCAT Slave configuration dialog opens.

➧ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set number of Input Data Bytes and Output Data Bytes

➢ Enter in the field Input Data Bytes the number of bytes the EtherCAT Master should send to the netTAP or netBRICK device, e. g. 32

➢ Enter in the field Output Data Bytes the number of bytes the EtherCAT Master should receive from the netTAP or netBRICK device, e. g. 64

3. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *EtherCAT Slave Parameter* on page 55.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the EtherCAT network as well as the used devices parameters needs to be adjusted if necessary.

4. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

5. Signal configuration

➢ If you want to assign own signal names, then enter your signal names.. Further information is described in section *Signal Configuration* on page 140.

6. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.7.1 EtherCAT Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex) Primary Hilscher Vendor ID: 0x00000044 (hex) Default: Secondary Hilscher Vendor ID: 0xE0000044 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NT 100-RE/ECS: 0x0000000D (hex), NB 100-RE/ECS: 0x00000019 (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: 0x00000000 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Data		
Input Data Bytes	Length of the input data in Byte	0 … 200 Byte Default: 4 Byte Default in XML-Datei: 200
Output Data Bytes	Length of the output data in Byte	0 … 200 Byte Default: 4 Byte Default in XML-Datei: 200
* Sum: The maximum length of input and of output is equal 400 Bytes,		

Table 13: EtherCAT Slave Parameters

© Hilscher, 2009-2010

4.7.2 Settings at the used EtherCAT Master

Device Description File

To configure the Master an XML file (device description file) is required. The CML file is on the delivered DVD in the directory EDS/EtherCAT:

Device	XML File
NT 100-RE-XX	Hilscher NT 100-ECS-XX V2.2.xml
NB 100-RE-XX	Hilscher NB 100-ECS-XX V2.2.xml

Table 14: XML File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Master must comply with the settings in the Slave, to establish a communication. Important parameters are: Vendor ID, Product Code, Serial Number, Revision Number, Output and Input length.

In order that the EtherCAT Master can communicate with the netTAP or netBRICK device as EtherCAT Slave:

- Enter at the EtherCAT Master the number of data bytes (number of input data), that are configured in the netTAP or netBRICK device, e. g. 64.
- Enter at the EtherCAT Master the number of data bytes (number of output data), that are configured in the netTAP or netBRICK device, e. g. 32.
- You can set at the EtherCAT Master, if the Master verifies identification numbers of the EtherCAT Slave. This verification can be activated or deactivated at the EtherCAT Master. If the verification is used, then use or check for the following values: vendor 68 (0x0044), product code 13 (0x000D) for NT 100 respectively product code 25 (0x00019) for NB 100, revision number 0 (0x0000).

© Hilscher, 2009-2010

4.8 Configuration EtherNet/IP Scanner

The netTAP NT 100 device respectively the netBRICK NB 100 device as Ethernet/IP Scanner needs a configuration, e. g. information about how many Ethernet/IP Adapter devices with how many input and output data are to be connected.

1. Insert Ethernet/IP Adapter devices

- Drag from the device catalog Ethernet/IP Adapter device(s) and drop it to the bus line of the Ethernet/IP Scanner.

2. Configure Ethernet/IP Adapter devices

- Open the configuration dialog for each Ethernet/IP Adapter device and configure the device.

  Information about how to configure the Ethernet/IP Adapter device is in the operating instruction manual „Generic DTM for EtherNet/IP Adapter devices“ and has the file name EtherNetIP_GenericAdapter_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure Ethernet/IP Scanner

- Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > Ethernet/IP Scanner**

  Information about how to configure the Ethernet/IP Scanner device is in the operating instruction manual „DTM for EtherNet/IP Scanner devices“ and has the file name EtherNetIP_Scanner_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.9 Configuration EtherNet/IP Adapter

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as EtherNet/IP Adapter needs parameter. The device can either work as a Client or as a Server.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > EtherNet/IP Adapter**

↬ The EtherNet/IP Adapter configuration dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the settings dialog is displayed.

2. EtherNet/IP Adapter Parameter

   To set a fix IP address:

➢ Deselect the flag DHCP and BootP.

➢ Select IP address Enable and enter a valid IP address for the netTAP or netBRICK device.

➢ Select netmask enable and enter a valid netmask.

➢ If needed: Select the gateway enable and enter a valid IP address for the gateway.

3. Set length produced and consumed data length

➢ Enter in the field produced data length the number of data bytes the EtherNet/IP Scanner should receive from the netTAP or netBRICK device, e.g. 64 bytes

➢ Enter in the field consumed data length the number of data bytes the EtherNet/IP Scanner should send to the netTAP or netBRICK device, e.g. 32 bytes

4. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *EtherNet/IP Adapter Parameter* on page 60.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the EtherNet/IP network as well as the used devices parameters needs to be adjusted if necessary.

© Hilscher, 2009-2010

5. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

6. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

7. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.9.1 EtherNet/IP Adapter Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000011B (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0x0000FFFF (hex), Default NT 50-RE/EIS: 0x000000113 (hex), NT 100-RE/EIS: 0x00000010F (hex), NB 100-RE/EIS: 0x000000111 (hex)
Product Type	Communication Adapter	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
Major Rev	Major Revision	0 ... 255, Default: 1
Minor Rev	Minor Revision	0 ... 255, Default: 1
Device name	Device name of the device station as character string, e. g. EtherNet/IP Adapter (Slave).	0 - 31 ASCII characters
Bus		
IP Address	Valid IP Address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its IP Address from a DHCP server or also from a BOOTP server, if this one is checked. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid IP address Default: unchecked
Netmask	Valid Network mask for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its Netmask from a DHCP server or also from a BOOTP server, if this one is checked. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid network mask Default: unchecked

Table 15: EtherNet/IP Adapter Parameters (Part 1)

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value/Value
Gateway	Valid Gateway Address for the device If 'Enabled' is unchecked (Default setting), the device obtains its Gateway Address from a DHCP server or also from a BOOTP server, if this one is checked. If 'Enabled' is checked, the device uses the manually entered value.	Valid gateway address Default: unchecked
	There are three methods available, how the device can obtain its IP Address, Netmask and Gateway Address, one of which must be selected. These methods can also be combined. The device performs the following sequence in order to obtain the addresses: 1. from a DHCP server if DHCP is checked (if a DHCP server provides the requested addresses to the device, then the device uses these addresses) 2. from a BootP server if BootP is checked (if a BootP server provides the requested addresses to the device, then the device uses these addresses) 3. the addresses manually set are used. If the IP Address is set manually also the Network Mask must be set manually. The manually set Gateway Address is optional. If no DHCP server and no BootP server and no manually set addresses exist, then the protocol is not ready for initialization or for operation.	
Flags	BootP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server.	Default: unchecked
	DHCP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a DHCP server.	Default: checked
	100Mbit: Speed Selection, If checked, the device will operate at 100 MBit/s, else at 10 MBit/s. This parameter will not be in effect, when auto-negotiation is active.	Default: unchecked
	FullDuplex: Duplex Operation, If checked, full-duplex operation will be used. The device will operate in half-duplex mode, if this parameter is set to zero. This parameter will not be in effect, when auto-negotiation is active.	Default: unchecked
	Auto-neg.: Auto-Negotiation, If checked, the device will auto-negotiate link parameters with the remote hub or switch.	Default: checked
Data		
Input Data Bytes	Maximum allowed length of the input data in Byte. This parameter should be equal to or higher than the complete projected input data length, otherwise the EtherNet/IP device will reject the cyclic communication requests.	0 … 504 Byte Default: 16 Byte
Output Data Bytes	Maximum allowed length of the output data in Byte. This parameter should be equal to or higher than the complete projected output data length, otherwise the EtherNet/IP device will reject the cyclic communication requests.	0 … 504 Byte Default: 16 Byte

Table 16: EtherNet/IP Adapter Parameters (Part 2)

© Hilscher, 2009-2010

4.9.2 Settings at the used EtherNet/IP Scanner

Device Description File

To configure the Scanner an EDS file (device description file) is required. The EDS file is on the delivered DVD in the directory EDS/EtherNetIP:

Device	EDS File
NT 50-XX-EN or NT 50-EN-XX	HILSCHER NT 50-RE EIS V1.1.EDS
NT 100-RE-XX	HILSCHER NT 100-RE EIS V1.1.EDS
NB 100-RE-XX	HILSCHER NB 100-RE EIS V1.1.EDS

Table 17: EDS File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Scanner must comply with the settings in the Adapter, to establish a communication. Important parameters are: Input, Output length, Vendor ID, Product Type, Product Code, Major Rev, Minor Rev, IP Address and Netmask.

In order that the EtherNet/IP Scanner can communicate with the netTAP or netBRICK device as EtherNet/IP Adapter:

- Enter at the EtherNet/IP Scanner the IP address of the netTAP or netBRICK device.
- Use at the EtherNet/IP Scanner the instance ID 101, to receive data from the netTAP or netBRICK device.
- Enter at the EtherNet/IP Scanner for this instance ID (101) the number of data (number of receive data bytes), which were configured in the netTAP or netBRICK device, e. g. 64.

Note: Some EtherNet/IP Scanners need for setting this value (number of receive data) a value increased by 4 (this is the length of the Run/Idle header, that can be transferred in front of the user data). With the example of 64 bytes above therefore 68 (64 + 4) needs to be set for the number of data.

- If adjustable at the EtherNet/IP Scanner, set that the EtherNet/IP Scanner sends the 32-Bit Run/Idle header.
- Use at the EtherNet/IP Scanner the instance ID 100, to send data to the netTAP or netBRICK device.
- Enter at the EtherNet/IP Scanner for this instance ID (100) the number of data (number of send data bytes), which were configured in the netTAP or netBRICK device, e. g. 32.
- If adjustable at the EtherNet/IP Scanner, set that the EtherNet/IP Scanner receives the 32-Bit Run/Idle header.
- You can set at the EtherNet/IP Scanner, if the Scanner verifies identification numbers of the EtherNet/IP Adapter (named keying). This verification can be activated or deactivated at the EtherNet/IP Scanner. If the verification is used, then use or check for the following values for the NT 100 device: vendor 283 (0x011B), product code 275 (0x0113) for NT 50 or product code 271 (0x010F) for NT 100 product code 273 (0x0111) for NB 100, product type 12 (0x000C), major revision 1, minor revision 1.

© Hilscher, 2009-2010

4.10 Configuration Open Modbus/TCP

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as Open Modbus/TCP needs parameter. The device can either work as a Client or as a Server.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > Open Modbus/TCP**

➧ The Open Modbus/TCP settings dialog opens.

➧ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the settings dialog is displayed.

2. Set protocol mode

➢ Set the **Protocol Mode** to "**I/O Server**", if the netTAP or netBRICK device should work as Open Modbus/TCP Server

➢ Set for the **Protocol Mode** to "**Client**", if the netTAP or netBRICK device should work as Open Modbus/TCP Client

➧ If you have set the **Protocol Mode** to "**Client**", then the entry **Command Table** appears in the navigation are

3. Open Modbus/TCP Parameter

   To set a fix IP address:

➢ Deselect the flag DHCP and BootP.

➢ Select IP address Enable and enter a valid IP address for the NT 100 device.

➢ Select netmask enable and enter a valid netmask.

➢ If needed: Select the gateway enable and enter a valid IP address for the gateway.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the Ethernet network as well as the used devices parameters needs to be adjusted if necessary. Further information is described in section Open Modbus/TCP Parameter on page 65.

© Hilscher, 2009-2010

4. Command Table

➢ If you have set the **Protocol Mode** to "**Client**", then select in the navigation area the entry **Command Table**. Specify in the Command Table which data the Open Modbus/TCP Client has to transfer to and from the Open Modbus/TCP Server by reading or writing. The Command Table is described in detail in section *Command Table Open Modbus/TCP* on page 67.

Note: To know, which data address with which data of the Open Modbus/TCP Server devices is provided for reading or for writing, refer to the device description of the Open Modbus/TCP Server device manufacturer.

5. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

↬ The signal configuration dialog opens

6. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

7. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

↬ The configuration dialog closes

© Hilscher, 2009-2010

4.10.1 Open Modbus/TCP Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication start automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
Protocolmode	Protocolmode: Client or I/O Server For „Client“ **Command Table** is displayed in the navigation area. For the Open Modbus/TCP Client the Command Table is a list with commands for the reading or writing of data.	I/O Server (default), Client
Data Swap	Data-storage mode: No: Data will not be swapped Yes: Data will be swapped.	Data will be swapped (default), Data will not be swapped
Map FC1 and FC3	If unchecked, data are read from the intput area using FC1, FC3 and FC23. If checked, data are read from the output area using FC1, FC3 and FC23. FC1 then can be used instead of FC2 and FC3 instead of FC4.	Default: unchecked/no alternative mapping
Bus		
Provided Server Connections	Server Connections Number of sockets to provide for server requests* *A value of 0 means that the Open Modbus/TCP task exclusive works as Client, while a Value of 16 means that the Open Modbus/TCP task exclusive works as Server in Message-Mode. The parameters Send Timeout, Connect Timeout and Close Timeout are for the Timeout between the Open Modbus/TCP Task and the TCP Task.	0 ... **4** ... 16
Client Connection Watchdog Time	Connection remain open time Only for client jobs in message-mode. The connection to the destination-device stays open, until timeout is expired. Value is multiplied with 100 ms. Note: This timeout starts, after receiving the answer to a command	100 ... **1000** ... 6000000 Default: NIC 50-RE/OMB: 1000
Response Timeout	Telegram Timeout Only for client jobs in message-mode. After expiration of this time, the job will be canceled and an error is send to the application. Value is multiplied with 100 ms. Note: This timeout starts after command is send to the destination device via TCP	100 ... **2000** ... 6000000 Default NIC 50-RE/OMB: 2000
Send Acknowledge-ment Timeout	TCP Task SendTimeout Parameter Parameter for TCP task (in milliseconds) . Used OMB task internal. It specifies the timeout for trying to send messages via TCP/IP If the value 0 is selected, the default value of 31000 milliseconds is used.	**0** ... 2.000.000.000
Connect Acknowledge-ment Timeout	TCP Task Connect Timeout Parameter Parameter for TCP task (in milliseconds). Used OMB task internal. It specifies the timeout for trying to establish a connection with the TCP task. If the value 0 is selected, the default value of 31000 milliseconds is used.	**0** ... 2.000.000.000
Close Acknowledge-ment Timeout	TCP Task Close Timeout Parameter Parameter for TCP task (in milliseconds). Used OMB task internal. It specifies the timeout for trying to close a connection with the TCP task. If the value 0 is selected, the default value of 13000 milliseconds is used.	**0** ... 2.000.000.000
IP Address	Valid IP address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its IP Address from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid IP address Default: unchecked
Net Mask	Valid Network mask for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its Netmask from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid network mask Default: unchecked

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value/Value
Gateway	Valid Gateway address for the device If ‘Enabled’ is unchecked (Default setting), the device obtains its Gateway Address from a DHCP or BOOTP server. If ‘Enabled’ is checked, the device uses the manually entered value.	Valid gateway address Default: unchecked
	There are three methods available, how the device can obtain its IP Address, Netmask and Gateway Address, one of which must be selected. These methods can also be combined. The device performs the following sequence in order to obtain the addresses: 1. from a DHCP server if DHCP is checked (if a DHCP server provides the requested addresses to the device, then the device uses these addresses) 2. from a BootP server if BootP is checked (if a BootP server provides the requested addresses to the device, then the device uses these addresses) 3. the addresses manually set are used. If the IP Address is set manually also the Network Mask must be set manually. The manually set Gateway Address is optional. If no DHCP server and no BootP server and no manually set addresses exist, then the protocol is not ready for initialization or for operation.	
Extras	BootP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a BOOTP server.	Default: unchecked.
	DHCP: If checked, the device obtains its IP Address, Netmask, Gateway Address from a DHCP server.	Default: unchecked.

Table 18: Open Modbus/TCP Parameters

© Hilscher, 2009-2010

4.10.2 Command Table Open Modbus/TCP

4.10.2.1 For what the Command Table does serve?

For the Open Modbus/TCP Client the **Command Table** is a list with commands for the reading or writing of data.

The **Command Table** is only relevant, if a device works as Client on the Open Modbus/TCP.

From every command line the Open Modbus/TCP Client produces a telegram for reading or for the writing of data to or from an Open Modbus/TCP Server device. For every command are indicated:

- the Open Modbus/TCP Server device address (Device Address),
- the Unit identifier (to identify a remote Server via gateway),
- the Function code (Function Code),
- the data address in the Open Modbus/TCP Server device (Address),
- the number of data (Number of Registers/Coils),
- and the data address in the Open Modbus/TCP Client device (Dual-Port Memory Address).

For writing telegrams (FC 5, 6, 15 and 16) in the column

- Trigger

you can decide, whether the writing telegrams are executed every time (Cyclic) or only at data modification (Change data).

Furthermore

- a Cycle time (Cycle Time)

can be set for every command.

The **Command Table** is processed from the first to the last entry (from above to below). After the execution of the last command the execution of the first command is started again.

For reading commands the Open Modbus/TCP Client reads out data from the Open Modbus/TCP Server and saves them into its data memory.

For writing commands the Open Modbus/TCP Client reads out data from its data memory and writes them into the Open Modbus/TCP Server.

The number of commands which can be defined depends from the firmware and from the Dual-Port Memory layout.

For Open Modbus one connection per IP address is established. In maximum 16 Client connections can be supported. In this case the parameter Open Server Sockets must be set from 4 (Default) to 0.

© Hilscher, 2009-2010

4.10.2.2 Parameter of the Command Table

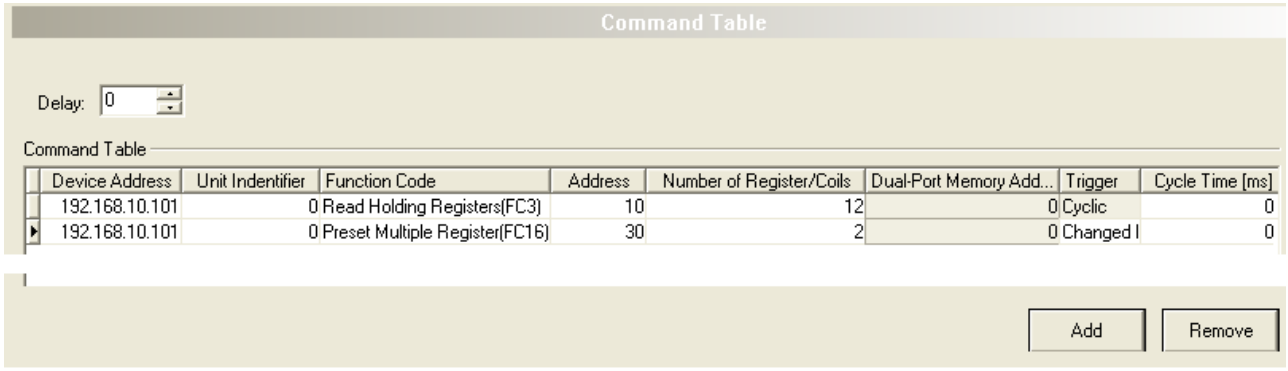


Figure 17: Pane Command Table

Each command contains the following parameters:

Parameter	Description
Device Address	Open Modbus/TCP Server IP address
Unit Identifier	to identify a remote Server via gateway
Function Code	Function code
Address	data address in the Open Modbus/TCP Server device
Number of Registers/Coils	Number of data
Dual-Port Memory Address	Data address in the Open Modbus/TCP Client device (= Byte address in the process image of the Open Modbus/TCP Client device)
Trigger	Trigger for writing commands
Cycle Time	Cycle time

Table 19: Parameter of the Command Table

© Hilscher, 2009-2010

<table>
<tr><th>Parameter</th><th>Meaning</th><th>Range of Value/Value</th></tr>
<tr><td>Device Address</td><td>Specifies the Open Modbus/TCP Server device address, from which the Open Modbus/TCP Client device reads out the data or into which it writes the data.</td><td>For Open Modbus/TCP:
000.000.000.000-
255.255.255.255,
Default: 000.000.000.000</td></tr>
<tr><td>Unit Identifier</td><td>To identify a remote Server connected on a serial line or on other buses.
The Unit Identifier is initialized by the Client. Do not change for response!</td><td>0 … 247,
Default: 0</td></tr>
<tr><td>Function Code</td><td>Indicates the function code for reading or writing commands [3]:
Read Coils(FC1)
Read Inputs(FC2)
Read Holding Registers(FC3)
Read Input Registers(FC4)
Force Single Coil(FC5)
Preset Single Register(FC6)
Force Multiple Coils(FC15)
Preset Multiple Registers(FC16)</td><td>1 / 2 / 3 / 4 / 5 / 6 / 15 / 16
Default: „Read Coils (FC1)“</td></tr>
<tr><td>Address</td><td>Indicates the data address in the Open Modbus/TCP Server device.
The parameter Address contains the register address in the Open Modbus/TCP Server device. The register address is counted for every function code beginning with 0.
The following table shows the assignment of the Address to the data address in the Open Modbus/TCP Server by Function code:
<table>
<tr><th rowspan="2">Parameter
Address</th><th colspan="4">Data Address Open Modbus/TCP Server</th></tr>
<tr><th>FC 1
FC 5
FC 15</th><th>FC 2</th><th>FC 4</th><th>FC 3
FC 6
FC 16</th></tr>
<tr><td>0</td><td>1</td><td>10001</td><td>30001</td><td>40001</td></tr>
<tr><td>1</td><td>2</td><td>10002</td><td>30002</td><td>40002</td></tr>
<tr><td>2</td><td>3</td><td>10003</td><td>30003</td><td>40003</td></tr>
<tr><td>3</td><td>4</td><td>10004</td><td>30004</td><td>40004</td></tr>
<tr><td>4</td><td>5</td><td>10005</td><td>30005</td><td>40005</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td></tr>
</table>

To know, which data address with which data of the Open Modbus/TCP Server devices is provided for reading or for writing, refer to the device description of the Open Modbus/TCP Server device manufacturer.</td><td></td></tr>
<tr><td>Number of Registers/Coils</td><td>Indicates the number of reading or writing data as registers or coils.
The maximum value depends from the function code.</td><td>1 … max. value, Default: 1
max. value for
FC1 = 2000 (Coils),
FC2 = 2000 (Coils),
FC3 = 125 (Registers),
FC4 = 125 (Registers),
FC5 = 1 (Coils),
FC6 = 1 (Registers),
FC15 = 1968 (Coils),
FC16 = 123 (Registers)</td></tr>
</table>

For more see next page

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value/Value
Dual-Port Memory Address	Byte address in the process image of the Open Modbus/TCP Client device for input or output data of the Client The configuration software calculates the byte address in the process image of the Client for the input or for the output data automatically. Because of this definition, for function codes for reading (FC 1, 2, 3 and 4) the data are assigned consecutively in the process image for input data and for function codes for writing (FC 5, 6, 15 and 16) the data are assigned consecutively in the process image for output data.	0 … 5759 Default: 0
Trigger	Defines for all function codes for writing (FC 5, 6, 15 and 16); if the command is to be executed cyclically (Cyclic) or only if the data have changed (Change Data). For function codes for reading (FC 1, 2, 3 und 4) this parameter is not used.	0 = Cyclic 1 = Changed Data Default: „Cyclic“
Cycle Time	The cycle time defines, after how many milliseconds a writing or a reading command shall be reexecuted. The default value „0 ms“ means, that the command is executed as fast as possible. Otherwise the execution can be defined in steps of 10 ms. With the cycle time the temporal execution of the command can be influenced. If since the last execution of the command the set cycle time or more time has been passed, the command is reexecuted. If since the last execution of the command less time has been passed then the cycle time, the command is skipped. *If e. g. from a Open Modbus/TCP Server data are required only every 10 seconds, as these one change only slowly, then for the cycle time enter 10.000 ms.*	0, 10, 20, 30, … 60.000 ms Default: 0 ms
Delay	Between the single commands a delay time can be parameterized. This sometimes is necessary to avoid a too high load of the connected Server devices by a continuous communication. **0**: The commands are processed without delay. **1 … 60.000 ms**: Delay time in ms, the Open Modbus/TCP Client waits, before starting the next command.	0 … 60.000 ms Default: 0

Table 20: Parameter of the Command Table

© Hilscher, 2009-2010

4.10.2.3 Adding/removing Command

To add or to remove data sets for commands to the Command Table, proceed as follows:

Add command:

- In the table **Command Table** put the cursor in the line below which a new data set for a command shall be added.
- Click **Add**.

↬ Under the clicked line a new command is added.

Remove command:

- In the table **Command Table** put the cursor in the line with the data set to be deleted.
- Click **Remove**.

↬ The command of the clicked line is not shown any more.

4.10.2.4 Setting Parameters

Important: When entering the parameters, make sure that these address valid data registers. (*Refer to the device description of the Open Modbus/TCP Server device manufacturer.*)

To set the single parameters in the Command Table, proceed as follows:

1. **Device Address**, **Function Code**, **Address**, **Number of Registers/Coils**:

The fields for these parameters are editable.

- Enter the values for these parameters to the corresponding table cells.

2. **Trigger**:

- In **Function Code** field select an entry for a writing command (FC 5, 6, 15 or 16).
- In the column **Trigger** select the entry „Cyclic“ or „Change Data“.

3. **Cycle Time**:

- Put the cursor to the table cell and set the cycle time using the spin box in steps of 10 ms.

© Hilscher, 2009-2010

4.10.2.5 Examples Open Modbus/TCP Reading or Writing Command

Device Address	Unit Indentifier	Function Code	Address	Number of Register/Coils	Dual-Port Memory Add...	Trigger	Cycle Time [ms]
192.168.10.101	0	Read Holding Registers(FC3)	10	12	0	Cyclic	0
192.168.10.101	0	Preset Multiple Register(FC16)	30	2	0	Changed Data	0

Figure 18: Examples - Reading Command with FC 3, Writing Command with FC16

Example Reading Command with FC 3: From the Open Modbus/TCP Server device with the Modbus address 8 from data address 40011 onwards 12 registers are read by function code 3. The data are assigned to the dual-port memory address 0.

Example Writing Command with FC 16: To the Open Modbus/TCP Server with the Modbus address 8 from data address 40031 onwards 12 registers are written by function code 16. The data are read from the process image of the Client from the dual-port memory address 0 onwards. The writing process is only performed, if the data in the process image of the Client on the dual-port memory addresses 0 to 3 have changed, as the parameter trigger is set to 'changed'.

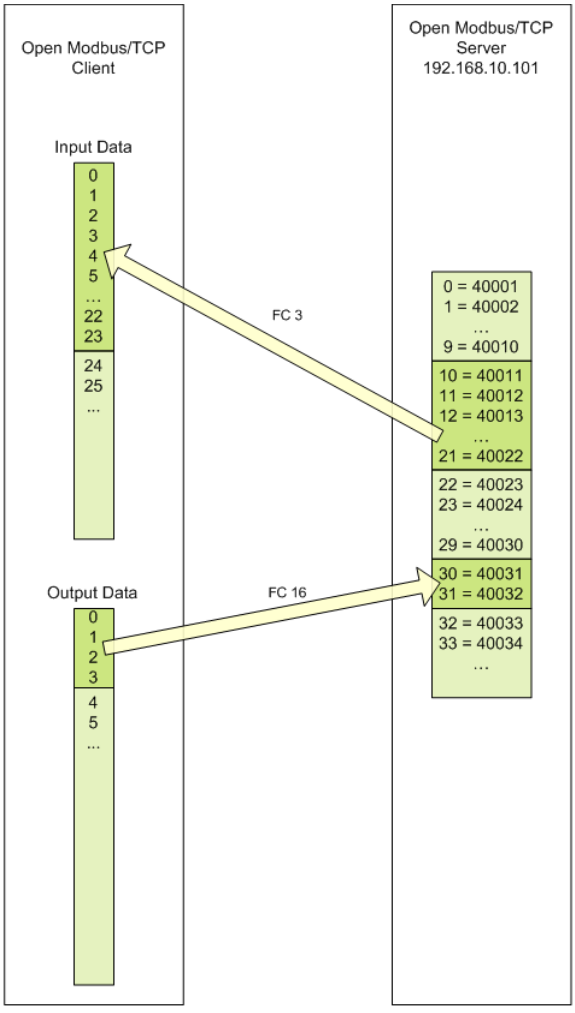


Figure 19: Examples - Reading Command with FC 3, Writing Command with FC16

© Hilscher, 2009-2010

4.10.3 Open Modbus/TCP Server

4.10.4 Settings at the used Open Modbus/TCP Client

In order that the Open Modbus/TCP Client can communicate with the netTAP or netBRICK device as an Open Modbus/TCP Server:

- Enter at the Open Modbus/TCP Client the IP address of the netTAP or netBRICK device
- Use at the Open Modbus/TCP Client port number 502 an. Port 502 is used by the netTAP or netBRICK device to receive and send Open Modbus telegrams.
- Use at the Open Modbus/TCP Client function code 4, to read data from the netTAP or netBRICK device. Data address 30001 to 32880
- Use at the Open Modbus/TCP Client function code 16, to write data into the netTAP or netBRICK device. Data address 40001 to 42880
- Use at the Open Modbus/TCP Client function code 3, to read data from the netTAP or netBRICK device. Data address 40001 to 42880

The Open Modbus/TCP Client can access to the netTAP or netBRICK device as an Open Modbus/TCP Server using function codes 1, 2, 3, 4, 5, 6, 15 and 16

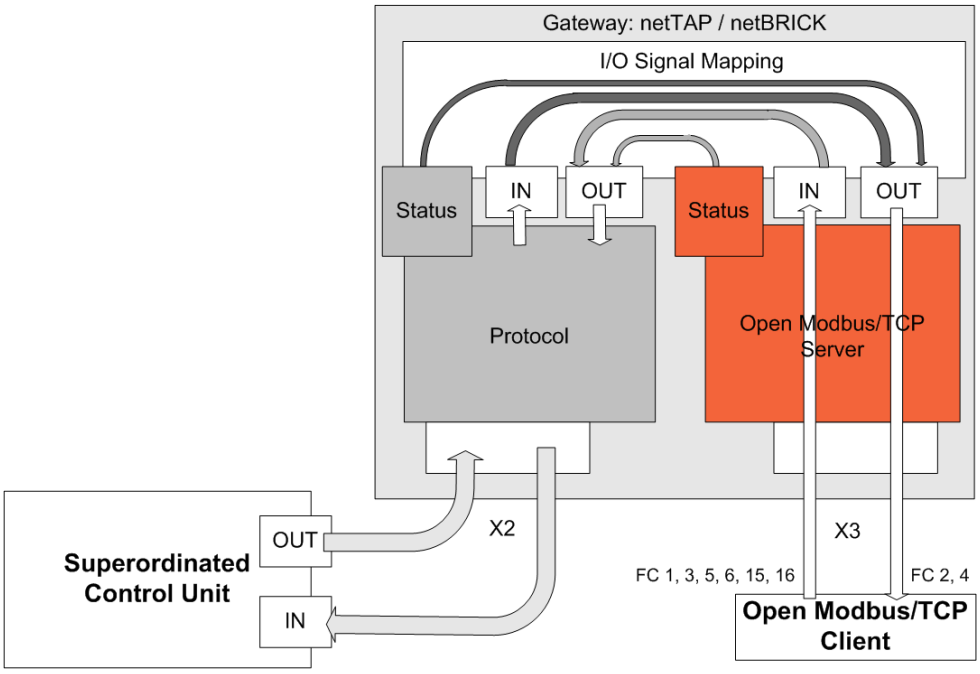


Figure 20: Access of the Open Modbus/TCP Client to the netTAP and netBRICK Device (1)

© Hilscher, 2009-2010

The following table shows the register and coil addresses for many typical Open Modbus/TCP Client systems:

Register or Coil of netTAP or netBRICK (Mapping)	Coil FC 1, FC 5, FC 15	Register FC 3, FC 6, FC 16
IN 0	1 … 16	40001
IN 1	17 … 32	40002
IN 2	33 … 48	40003
…	…	…

Register or Coil of netTAP or netBRICK (Mapping)	Coil FC 2	Register FC 4
OUT 0	10001 … 10016	30001
OUT 1	10017 … 10032	30002
OUT 2	10033 … 10048	30003
…	…	…

Figure 21: Access of the Open Modbus/TCP Client to the netTAP and netBRICK Device (2)

Note: Typical Open Modbus/TCP Client systems address on application level the first register with 40001 for function code 3 and 16. However, some Open Modbus/TCP Client systems address on application level the first register with 0, others address the first register with 1. Read the documentation of the used Open Modbus/TCP Client system to find out the used addressing of this system.

4.10.5 Modbus References

[1] MODBUS Application Protocol Specification V1.1, http://www.modbus.org/, 12/06/02

[2] MODBUS Messaging on TCP/IP Implementation Guide V1.0b, October 24, 2006

[3] MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1a, June 4, 2004, http://www.Modbus-IDA.org

© Hilscher, 2009-2010

4.11 Configuration POWERLINK Controlled Node

The netTAP NT 100 device respectively the netBRICK NB 100 device as POWERLINK Controlled Node needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > POWERLINK Controlled Node**

↬ The POWERLINK Controlled Node configuration dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set Node-ID

➢ Enter in the field **Node-ID** the address, which the netTAP or netBRICK device uses on the POWERLINK network to be addressed by the POWERLINK Managing Node.

3. Set input data length and output data length

➢ Set in the field **Input Data Length** the number of data bytes the netTAP or netBRICK device should receive from the POWERLINK Managing Node

➢ Set in the field **Output Data Length** the number of data bytes the netTAP or netBRICK device should send to the POWERLINK Managing Node

4. Enable PDO Mapping Version Check

➢ Uncheck the field **Disable PDO Mapping Version Check**

5. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *POWERLINK Controlled Node Parameter* on page 77.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the POWERLINK network as well as the used devices parameters needs to be adjusted if necessary.

© Hilscher, 2009-2010

6. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

7. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

8. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.11.1 POWERLINK Controlled Node Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Disable Host-Triggered Input Data Exchange	Decides, whether host-triggered update for input data exchange is enabled (hook checked) or disabled (hook unchecked).	Default: unchecked
Disable Host-Triggered Output Data Exchange	Decides, whether host-triggered update for output data exchange is enabled (hook checked) or disabled (hook unchecked).	Default: unchecked
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x00000044 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex), Default 0x00000000 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NT 100-RE/PLS: 0x00000018 (hex), NB 100-RE/PLS: 0x00000017 (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default NT 100-RE/PLS, NB 100-RE/PLS: 0x00000000 (hex)
Bus		
Node Id	EPL Node ID (EPL = Ethernet Powerlink)	1...239, Default: 1
DNS Node Name	DNS-compatible name of the Powerlink Controlled Node/Slave optional	
Gateway Address	Gateway address for IP stack	Default: 192.168.100.254

Table 21: POWERLINK Controlled Node/Slave Parameters (Part 1)

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value/Value
Data		
Input Data Bytes	Length of the input data in byte	1… 1490 Byte, Default: 4 Byte
Output Data Bytes	Length of the output data in byte	1… 1490 Byte, Default: 4 Byte
Disable PDO Mapping Version Check	Decides, whether PDO mapping version is checked (hook checked) or not checked (hook unchecked). **Select for netTAP and netBRICK devices always unchecked.**	Default: checked
Configure Default Objects	Decides, whether default objects are created (hook checked) or not (hook unchecked). If the objects will be created, the old set of previously existing objects will be cleared	Default: checked
Delete Application Objects	Decides, whether application objects are deleted (hook checked) or not (hook unchecked).	Default: checked

Table 22: POWERLINK Controlled Node/Slave Parameters (Part 2)

© Hilscher, 2009-2010

4.11.2 Settings at the used POWERLINK Managing Node

Device Description File

To configure the Managing Node a XDD file (device description file) is required. The XDD file is on the delivered DVD in the directory EDS/POWERLINK:

Device	XDD File
NT 100-RE-XX	00000044_NT 100 RE PLS.XDD
NB 100-RE-XX	00000044_NB 100 RE PLS.XDD

Table 23: XDD File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Managing Node must comply with the settings in the Controlled Node, to establish communication. Important parameters are: Vendor ID, Product Code, Serial Number, Revision Number, Node ID, Output and Input length.

In order that the POWERLINK Managing Node can communicate with the netTAP or netBRICK device as POWERLINK Controlled Node:

- Enter at the POWERLINK Managing Node the Node-ID, which is configured in the netTAP or netBRICK device.
- Enter for the length of output data in the POWERLINK Managing Node as set in the netTAP or netBRICK device for the Output Data Bytes.
- Enter for the length of input data in the POWERLINK Managing Node as set in the netTAP or netBRICK device for the Input Data Bytes.
- The POWERLINK Managing Node verifies identification numbers of the netTAP or netBRICK device as POWERLINK Controlled Node. Use or check for the following values for the netTAP or netBRICK device: vendor 68 (0x44), serial number 0, product code 24 (0x18) for NT 100 respectively product code 23 (0x17) for NB 100, revision number 0.

© Hilscher, 2009-2010

4.12 Configuration PROFINET IO Controller

The netTAP NT 100 device respectively the netBRICK NB 100 device as PROFINET IO Controller needs a configuration, e. g. information about how many PROFINET IO Device devices with how many input and output data are to be connected.

1. Insert PROFINET IO Device devices

➢ Drag from the device catalog PROFINET IO Device device(s) and drop it to the bus line of the PROFINET IO Controller.

2. Configure PROFINET IO Device devices

➢ Open the configuration dialog for each PROFINET IO Device device and configure the device.

Information about how to configure the PROFINET IO Device device is in the operating instruction manual „Generic DTM for PROFINET IO Device devices“ and has the file name PROFINET_IO_GenericDevice_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure PROFINET IO Controller

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > PROFINET IO Controller**

Information about how to configure the PROFINET IO Controller device is in the operating instruction manual „DTM for PROFINET IO Controller devices“ and has the file name PROFINET_IO_Controller_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.13 Configuration PROFINET IO Device (Gateway)

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as PROFINET IO Device needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > PROFINET IO Device**

↬ The PROFINET IO Device configuration dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the settings dialog is displayed.

2. Set the number of Input Data Bytes and Output Data Bytes

➢ Enter in the field Input Data Bytes the number of data bytes, which should be transferred from the PROFINET IO Controller to the netTAP or netBRICK device, e. g. 32

➢ Enter in the field Output Data Bytes the number of data bytes, which should be transferred from the netTAP or netBRICK device to the PROFINET IO Controller, e. g. 64

3. Station name

   The default name for the **Name of Station** for the NT 50 device is nt50enpns, for the NT 100 device nt100repns and for the NB 100 device nb100repns. If you need to set an other name of station or you use several netTAP or netBRICK devices in the PROFINET network, then set a unique name of station as follows:

➢ Select Enable in the Ident area.

➢ Enter in the field Name of Station the name of station, which the netTAP or netBRICK device should use in the PROFINET network.

➢ Enter in the field Vendor ID the value 0x011E.

➢ Enter in the field Device ID for the NT 50 value 0x010F, for the NT 100 value 0x010B respectively for the NB 100 value 0x010E.

➢ Values in the fields Device Type, Order ID and Type of Station are optional.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the PROFINET network as well as the used devices parameters needs to be adjusted if necessary. Further information is described in section *PROFINET IO Device Parameter* on page 83.

© Hilscher, 2009-2010

4. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

↬ The signal configuration dialog opens

5. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

6. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

↬ The configuration dialog closes

© Hilscher, 2009-2010

4.13.1 PROFINET IO Device Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer, assigned by PROFIBUS Nutzerorganisation e. V.	0x00000000 ... 0xFFFFFFFF (hex), Hilscher: 0x00000011E (hex)
Device ID	Identification number of the device, freely eligible by the manufacturer, fixed for every device.	0x00000000 ... 0x0000FFFF (hex), NT 50-EN/PNS (Gateway): 0x0000010F (hex), NT 100-RE/PNS (Gateway) 0x00000010B (hex), NT 100_RE/PNS (Proxy) 0x000000106 (hex), NB 100-RE/PNS (Gateway) 0x00000010E (hex)
Device Type	Description of the device type, freely eligible	Character string, 0 ... 25 characters
Order ID	Hilscher device number or order description of the customer for its device	Character string, 0 ... 20 characters
Name of Station	Station name of the PROFINET IO-Device station. It has to match the station name configured in the PROFINET IO-Controller for this device. Must be DNS compatible name.	Character string, 1 ... 240 characters, Default: See GSDML-File
Type of Station	Type name of the PROFINET station; name can be assigned freely.	Character string, 1 ... 240 characters Default: See GSDML-File
Data		
Input Data Bytes	Maximum allowed length of the input data in Byte. This parameter should be equal to or higher than the complete projected input data length, otherwise the IO Device will reject the cyclic communication requests.	0 … 1024 Byte Default: 128 Byte
Output Data Bytes	Maximum allowed length of the output data in Byte. This parameter should be equal to or higher than the complete projected output data length, otherwise the IO Device will reject the cyclic communication requests.	0 … 1024 Byte Default: 128 Byte

Table 24: PROFINET IO-Device Parameters

© Hilscher, 2009-2010

4.13.2 Settings at the used PROFINET IO Controller

Device Description File

To configure the Master a GSDML file (device description file) is required. The GSDML file is on the delivered DVD in the directory EDS/PROFINET:

Device	GSDML File
NT 50-XX-EN or NT 50-EN-XX	GSDML-V2.1-HILSCHER-NT 50-EN PNS-20100219.xml
NT 100-RE-XX	GSDML-V2.1-HILSCHER-NT 100-RE PNS-20090123.xml
NB 100-RE-XX	GSDML-V2.1-HILSCHER-NB 100-RE PNS-20091112.xml

Table 25: GSDML File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Controller must comply with the settings in the Device, to establish a communication. Important parameters are: Name of Station, Vendor ID, Device ID, Input and Output data length.

In order that the PROFINET IO Controller can communicate with the netTAP or netBRICK device as PROFINET IO Device:

Note: Under **Name of Station** in the PROFINET IO Controller the name must be used, which was configured in the netTAP or netBRICK device.

- Enter at the PROFINET IO Controller the Name of Station, which is configured in the NT 100 device.
- Use for the configuration of the PROFINET IO Controller modules e. g. ‚32 Bytes In', for receiving data from the netTAP or netBRICK device. Several modules can also be used, e. g. 2 times the module ‚16 Bytes In'. The number of data (counted in bytes), that results from the used modules for input data, has to match with the number of input data configured in the netTAP or netBRICK device.
- Use for the configuration of the PROFINET IO Controller modules e. g. ‚64 Bytes Out', for sending data to the netTAP or netBRICK device. Several modules can also be used, e. g. 4 times the module ‚16 Bytes Out'. The number of data (counted in bytes), that results from the used modules for output data, has to match with the number of output data configured in the netTAP or netBRICK device.
- The PROFINET IO Controller verifies identification numbers of the netTAP respectively netBRICK device: Vendor ID (value 0x011E (286)), device type (value for NT 50 is 0x010F (271), value for NT 100 is 0x010B (267), value for NB 100 is 0x010E (270)). The PROFINET IO Controller gets these numbers from the GSDML file (name see above).

© Hilscher, 2009-2010

4.14 Configuration SERCOS III Slave

The NT 100 device respectively the netBRICK NB 100 device as SERCOS III Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > SERCOS III Slave**

➧ The SERCOS III Slave configuration dialog opens.

➧ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set Device Address

➢ Enter in the field Device Address the address, which the netTAP or netBRICK device uses on the SERCOS III network to be addressed by the SERCOS III Master.

4. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

5. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

6. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.14.1 SERCOS III Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Bus		
Device Address	Address for the SERCOS III Slave. The address range is from 1 to 127.	[1 … 127], Default: 1
IP Address Networkmask Gateway Flags (BootP, DHCP)	The SERCOS III Slave does not support to set the IP Address, Network Mask and Gateway Address or the transmission methods DHCP und BootP.	
Data		
Object dictionary	Location of the Object Dictionary for Service Channel: local or host Use for netTAP and netBRICK devices local only.	local, host 0= local 1= Host (not supported yet) Default = 0

Table 26: SERCOS III Slave Parameters

© Hilscher, 2009-2010

4.14.2 Settings at the used SERCOS III Master

Device Description File

To configure the Master an XML file (device description file) may be required. The XML file is on the delivered DVD in the directory EDS/SERCOS_III:

Device	XML File
NT 100-RE-XX	Hilscher NT 100-RE S3S.xml
NB 100-RE-XX	Hilscher NB 100-RE S3S.xml

Table 27: XML File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Device address.

© Hilscher, 2009-2010

4.15 Configuration CANopen Master

The netTAP NT 100 device respectively the netBRICK NB 100 device as CANopen Master needs a configuration, e. g. information about how many CANopen Slave devices with how many input and output data are to be connected.

1. Insert CANopen Slave devices

➢ Drag from the device catalog CANopen Slave device(s) and drop it to the bus line of the CANopen Master.

2. Configure CANopen Slave devices

➢ Open the configuration dialog for each CANopen Slave device and configure the device.

Information about how to configure the CANopen Slave device is in the operating instruction manual „Generic DTM for CANopen Slave devices“ and has the file name CANopen_Generic_Slave_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure CANopen Master

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > CANopen Master**

Information about how to configure the CANopen Master device is in the operating instruction manual „DTM for CANopen Master devices“ and has the file name CANopen_Master_netX_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.16 Configuration CANopen Slave

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as CANopen Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > CANopen Slave**

↬ The DeviceNet Slave configuration dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set Node-ID

➢ Enter in the field **Node-ID** the address, which the netTAP or netBRICK device uses on the CANopen network to be addressed by the CANopen Master.

3. Set baudrate

➢ Set the baudrate for the netTAP or netBRICK device, which is used in the CANopen network

4. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *CANopen Slave Parameter* on page 89.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the DeviceNet network as well as the used devices parameters needs to be adjusted if necessary.

Note: The maximum number of output data bytes and input data bytes is for the netTAP or netBRICK device 512 bytes each. The number of output data bytes and input data bytes is not adjustable in the Slave and therefore greyed out in the dialog.

The number of output data bytes and input data bytes, which are to be transferred effectively between the CANopen Master and Slave, are configured in the used CANopen Master. The CANopen Master configures the netTAP or netBRICK device during establishing the communication and set the number of output data bytes and input data bytes at this same time.

© Hilscher, 2009-2010

5. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

6. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

7. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.16.1 CANopen Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000044 (hex)
Product Code	Product code of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NT 100 CO/COS: 0x001A2020 (hex)
Revision Number	Revision number of the device as specified by the manufacturer	0x00000000 ... 0xFFFFFFFF (hex), Default: NT 50 CO/COS, NT 100 CO/COS, NB 100 CO/COS: 0x00020000 (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Bus		
Node Address	Node ID of CANopen slave	1 … 127, Default: NT 50 CO/COS, NT 100 CO/COS, NB 100 CO/COS: 2
Baudrate	Baud rate of CANopen connection	1 Mbaud, 800 Kbaud, 500 Kbaud, 250 Kbaud, 125 Kbaud, 100 Kbaud, 50 Kbaud, 20 Kbaud, 10 Kbaud, Default: NT 50 CO/COS, NT 100 CO/COS, NB 100 CO/COS: 1 MBaud

Table 28: CANopen Slave Parameters (Part 1)

© Hilscher, 2009-2010

Data		
Send Object /Receive Object	**Send Object**: Send object index **Receive Object**: Receive object index	0x00002000 ... 0x00002003 (hex) 0x00002200 ... 0x00002203 (hex)
	Size: Number of data Bytes to send per send object or number of data Bytes to receive per send object.	128
Output Data Bytes	Total output data Bytes of all send objects	Default: NT 50 CO/COS, NT 100 CO/COS, NB 100 CO/COS: 512 Bytes
Input Data Bytes	Total intput data Bytes of all receive objects	Default: NT 50 CO/COS, NT 100 CO/COS, NB 100 CO/COS: 512 Bytes

Table 29: CANopen Slave Parameters (Part 2)

Note: To configure the Master, an EDS file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Node Address and Baudrate.

© Hilscher, 2009-2010

4.16.2 Settings at the used CANopen Master

Device Description File

To configure the Master an EDS file (device description file) may be required. The EDS file is on the delivered DVD in the directory EDS/CANopen:

Device	EDS File
NT 50-XX-CO or NT 50-CO-XX	NT50_CO_COS.EDS
NT 100-XX-CO or NT 100-CO-XX	NT100_CO_COS.EDS
NB 100-XX-CO or NB 100-CO-XX	NB100_CO_COS.EDS

Table 30: EDS File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Node Address and Baudrate.

In order that the CANopen Master can communicate with the netTAP or netBRICK device as CANopen Slave:

- Enter at the CANopen Master the Node-ID, which is configured in the netTAP or netBRICK device.
- Configure at the CANopen Master all PDOs, which the Master should send to the netTAP or netBRICK device. For the netTAP or netBRICK device these are receive PDOs. E. g. the Master can be configured with 2 PDOs with 8 byte user data each (16 bytes in total) to be send to the netTAP or netBRICK device.
- Configure at the CANopen Master all PDOs, which the Master should receive from the netTAP or netBRICK device. For the netTAP or netBRICK device these are transmit PDOs. E. g. the Master can be configured with 4 PDOs with 8 byte user data each (32 bytes in total) to be received from the netTAP or netBRICK device.

© Hilscher, 2009-2010

4.17 Configuration CC-Link Slave

The netTAP NT 50 and NT 100 device as CC-Link Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP symbol the entry **Configuration > CC-Link Slave**

➢ The CC-Link Slave configuration dialog opens.

➢ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set Stationaddress

➢ Enter in the field **Stationaddress** the address, which the netTAP device uses on the CC-Link network to be addressed by the CC-Link Master.

3. Set baudrate

➢ Set the baudrate for the netTAP device, which is used in the CC-Link network

4. Set the CC-Link version

➢ Select with the field **CC-Link Version** the CC-Link version the netTAP device should use to communicate with the CC-Link Master.

5. Set the Station type

➢ If CC-Link Version 1 is to be used, then select with the field **Station Type** either **Remote Device Station** or **Remote IO Station** for the netTAP device.

6. Set the Number of Stations

➢ If Station type is **Remote Device Station**, then set the number of stations for the netTAP device: 1, 2, 3 or 4.

7. Set number of extension cycles

➢ If CC-Link Version 2 is to be used, then set the number of extension cycles for the netTAP device: 1, 2, 3 or 4.

Note: The number of IO-Data bytes depends on the following settings: station type, number of stations and number extension cycles. The number of stations can only be configured with station type Remote Device Station version 1 and version 2 and the number of extension cycles can only be configured with version 2.

© Hilscher, 2009-2010

8. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *CC-Link Slave Parameter* on page 96.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the CC-Link network as well as the used devices parameters needs to be adjusted if necessary.

9. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➢ The signal configuration dialog opens

10. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

11. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➢ The configuration dialog closes

© Hilscher, 2009-2010

4.17.1 CC-Link Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor Code	Code for the Vendor	0 ... 65535 resp. 0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000352 (hex)
Model Type	Model type	0 ... 255, Default: 1
Software Version	Software version	0 ... 63, Default: 0
Bus		
Station Address	Station address of CC-Link Slave Note: The number of occupied stations plus station address must not exceed the parameter range	1 ... 64
Baud Rate	Network transmission rate	156 kBaud (Default) 625 kBaud 2500 kBaud 5 MBaud 10 MBaud
Hold last received Output Data	Hold Clear Mode; Behavior in case of bus error Clear output data (unchecked) Hold last received output data (checked)	Default: unchecked

Table 31: CC-Link Slave Parameters (Part 1)

© Hilscher, 2009-2010

Data		
CC-Link Version	CC-Link Version 1 CC-Link Version 2	1 (Default) 2
Station Type	Type of CC-Link station Remote I/O Station: Remote Device Station	 0 (Default) 1
Number of Stations	Number of occupied stations Remote I/O Station: Remote Device Station:	 1 (Default) 1 ... 4
Extension Cycle	Number of extension cycles Allowed numbers for CC-Link version 1: Single/One cycle Allowed numbers for CC-Link version 2: Single/One cycle Double/Two cycles Quadruple/Four cycles Octuple/Eight cycles	 1 1 (Default) 2 4 8
IO-Data Bytes	The number of IO-Data bytes depends on the following settings: station type, number of stations and number extension cycles. The number of stations can only be configured with station type Remote Device Station version 1 and version 2 and the number of extension cycles can only be configured with version 2. Firmware/stack works according to CC-Link Version 2.0 input data output data Firmware/stack works according to CC-Link Version 1.11 input data output data	 12 … 368 Bytes 12 … 368 Bytes 4 … 48 Bytes 4 … 48 Bytes Default: 4

Table 32: CC-Link Slave Parameters (Part 2)

Note: To configure the Master, a CSP file (device description file) is required. The settings in the used Master must comply with the settings in the Slave to establish communication.

Important parameters are: Slave Station Address, Baudrate, Station Type and Vendor Code. For CC-Link Version 2.00 are additionally important: number of cycles as well as number of extension cycles.

© Hilscher, 2009-2010

4.17.2 Settings at the used CC-Link Master

Device Description File

To configure the Master, a CSP file (device description file) may be required. The CSP file is on the delivered DVD in the directory EDS/CC-Link:

Device	CSP File
NT 50-XX-CC or NT 50-CC-XX	nt50-cc-ccs_1.csp, nt50-cc-ccs_2.csp, nt50-cc-ccs_3.csp, nt50-cc-ccs_4.csp, nt50-cc-ccs_io.csp
NT 100-XX-CC or NT 100-CC-XX	Nt100-cc-ccs_1.csp, nt100-cc-ccs_2.csp, nt100-cc-ccs_3.csp, nt100-cc-ccs_4.csp, nt100-cc-ccs_io.csp

Table 33: CSP File Names for netTAP

The files nt50-cc-ccs_1.csp and nt100-cc-ccs_1.csp describe the respective netTAP device as one Remote Device Station, the files nt50-cc-ccs_2.csp and nt100-cc-ccs_2.csp describe the respective netTAP device as two Remote Device Station, etc. The files nt50-cc-ccs_io.csp and nt100-cc-ccs_io.csp describe the netTAP device as one Remote IO Station.

Configuration

Note: The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: Slave Station Address, Baudrate and Station Type. For CC-Link Version 1 are additionally important: Station type. For CC-Link Version 2.00 are additionally important: number of cycles as well as number of extension cycles.

In order that the CC-Link Master can communicate with the netTAP device as CC-Link Slave:

- Enter at the CC-Link Master the Stationaddress, which is configured in the netTAP device.
- Set at the CC-Link Master the same CC-Link Version as configured in the netTAP device.
- Set at the CC-Link Master the same station type as configured in the netTAP device.
- If CC-Link Version 1 is to be used, then set at the CC-Link Master the same Station type as configured in the netTAP device.
- If Station type is **Remote Device Station** to be used, then set at the CC-Link Master the same the number of stations as configured in the netTAP device.
- If CC-Link Version 2 is to be used, then set at the CC-Link Master the same number of extension as configured in the netTAP device.

© Hilscher, 2009-2010

4.18 Configuration DeviceNet Master

The netTAP NT 100 device respectively the netBRICK NB 100 device as DeviceNet Master needs a configuration, e. g. information about how many DeviceNet Slave devices with how many input and output data are to be connected.

1. Insert DeviceNet Slave devices

➢ Drag from the device catalog DeviceNet Slave device(s) and drop it to the bus line of the DeviceNet Master.

2. Configure DeviceNet Slave devices

➢ Open the configuration dialog for each DeviceNet Slave device and configure the device.

Information about how to configure the DeviceNet Slave device is in the operating instruction manual „Generic DTM for DeviceNet Slave devices“ and has the file name DeviceNet_Generic_Slave_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure DeviceNet Master

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > DeviceNet Master**

Information about how to configure the DeviceNet Master device is in the operating instruction manual „DTM for DeviceNet Master devices“ and has the file name DeviceNet_Master_netX_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.19 Configuration DeviceNet Slave

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as DeviceNet Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > DeviceNet Slave**

➢ The DeviceNet Slave configuration dialog opens.

➢ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set MAC-ID

➢ Enter in the field **MAC-ID** the address, which the netTAP or netBRICK device uses on the DeviceNet network to be addressed by the DeviceNet Master.

3. Set baudrate

➢ Set the baudrate for the netTAP or netBRICK device, which is used in the DeviceNet network

4. Set produced data length and consumed data length

➢ Set in the field **Prod. Data Length** the number of data bytes the netTAP or netBRICK device should send to the DeviceNet Master

➢ Set in the field **Cons. Data Length** the number of data bytes the netTAP or netBRICK device should receive from to the DeviceNet Master

© Hilscher, 2009-2010

5. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *DeviceNet Slave Parameter* on page 102.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the DeviceNet network as well as the used devices parameters needs to be adjusted if necessary.

6. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

↬ The signal configuration dialog opens

7. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

8. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

↬ The configuration dialog closes

© Hilscher, 2009-2010

4.19.1 DeviceNet Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication starts automatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Enabled	If ‘Enabled’ is unchecked, for the single Ident parameters each the respective default value is used.	
Vendor ID	Identification number of the manufacturer	0x00000000 ... 0x0000FFFF (hex), Hilscher: 0x00000011B (hex)
Product Code	Product code of the device	0x00000000 ... 0xFFFFFFFF (hex), Default NT 50 DN/DNS: 0x0000002D (hex), NT 100 DN/DNS: 0x00000002B (hex), NB 100 DN/DNS: 0x0000002F (hex)
Serial Number	Serial number of the device	0x00000000 ... 0xFFFFFFFF (hex)
Product Type	Communication Adapter	0x00000000 ... 0x0000FFFF (hex), Default: 0x00000000C (hex)
Minor Rev	Minor Revision	1 ... 255, Default: 1
Major Rev	Major Revision	1 ... 255, Default: 1
Product Name	The variable Product Name is a text string that should represent a short description of the product/product family.	0 ... 31 ASCII Characters

Table 34: DeviceNet Slave Parameters (Part 1)

© Hilscher, 2009-2010

Bus		
MAC ID	This parameter defines the DeviceNet address of the device within the network.	0 … 63, Default: 2
Baudrate	Baud rate of DeviceNet connection	500 kBaud, 250 kBaud, 125 kBaud, Default: 500 kBaud
Extras	Ignore address switch: address switches are always ignored	Default: unchecked
	Continue on CAN bus off: Unchecked: A device reset by the user is necessary in case of a CAN bus off event (e. g. short circuit of the data lines) Checked: The device tries independently to continue operation in case of a CAN bus off event	Default: unchecked
	Continue On Loss of Network Power (NP): Function not supported	Default: unchecked
	Receive-Idle Clear Data: Unchecked: Received data keep their last state in case of idle state Checked: Received data were set to zero in case of idle state	Default: unchecked
	Receive Idle keeps Data: Function not supported	Default: unchecked
Data		
Produced Size	ProducedSize sets the number of send bytes.	0 … 255, Default: 8
Consumed Size	ConsumedSize sets the number of receive bytes.	0 … 255, Default: 8

Table 35: DeviceNet Slave Parameters (Part 2)

© Hilscher, 2009-2010

4.19.2 Settings at the used DeviceNet Master

Device Description File

To configure the Master an EDS file (device description file) is required. The EDS file is on the delivered DVD in the directory EDS/DeviceNet:

Device	EDS File
NT 50-XX-DN or NT 50-DN-XX	NT50_DN_DNS.EDS
NT 100-XX-DN or NT 100-DN-XX	NT100_DN_DNS.EDS
NB 100-XX-DN or NB 100-DN-XX	NB100_DN_DNS.EDS

Table 36: EDS File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Master must comply with the settings in the Slave to establish communication. Important parameters are: MAC ID, Baudrate, Produced Size, Consumed Size, Vendor ID, Product Type, Product Code, Major Rev, Minor Rev.

In order that the DeviceNet Master can communicate with the netTAP or netBRICK device as DeviceNet Slave:

- Enter at the DeviceNet Master the MAC-ID, which is configured in the netTAP or netBRICK device.
- Enter for the length of output data in the DeviceNet Master the Cons. Data Length as set in the netTAP or netBRICK device.
- Enter for the length of input data in the DeviceNet Master the Prod. Data Length as set in the netTAP or netBRICK device.
- You can set at the DeviceNet Master, if the Master verifies identification numbers of the DeviceNet Slave (named keying). This verification can be activated or deactivated at the DeviceNet Master. If the verification is used, then use or check for the following values for the netTAP or netBRICK device: vendor 283 (0x011B), product code 45 (0x2D) for NT 50 respectively product code 43 (0x2B) for NT 100 respectively product code 47 (0x2F) for NB 100, product type 12 (0x000C), major revision 1, minor revision 1.

© Hilscher, 2009-2010

4.20 Configuration PROFIBUS-DP Master

The netTAP NT 100 device respectively the netBRICK NB 100 device as PROFIBUS-DP Master needs a configuration, e. g. information about how many PROFIBUS-DP Slave devices with how many input and output data are to be connected.

1. Insert PROFIBUS-DP Slave devices

➢ Drag from the device catalog PROFIBUS-DP Slave device(s) and drop it to the bus line of the PROFIBUS-DP Master.

2. Configure PROFIBUS-DP Slave devices

➢ Open the configuration dialog for each PROFIBUS-DP Slave device and configure the device.

Information about how to configure the PROFIBUS-DP Slave device is in the operating instruction manual „Generic DTM for PROFIBUS-DP Slave devices“ and has the file name PROFIBUS_Generic_Slave_DTM_en.pdf respectively open in the configuration dialog of the slave the online help by the F1 key.

3. Configure PROFIBUS-DP Master

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > PROFIBUS-DP Master**

Information about how to configure the PROFIBUS-DP Master device is in the operating instruction manual „DTM for PROFIBUS-DP Master devices“ and has the file name PROFIBUS_Master_netX_DTM_en.pdf respectively open in the configuration dialog of the master the online help by the F1 key.

© Hilscher, 2009-2010

4.21 Configuration PROFIBUS-DP Slave

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as PROFIBUS-DP Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > PROFIBUS-DP Slave**

↬ The PROFIBUS-DP Slave configuration dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the configuration dialog is displayed.

2. Set station address

➢ Enter in the field **Station address** the address, which the netTAP or netBRICK device uses on the PROFIBUS network to be addressed by the PROFIBUS-DP Master.

3. Set baudrate

➢ Use the setting Auto-Detect (default), if the netTAP or netBRICK device should detect the baudrate on the PROFIBUS network or set the baud rate for the device.

4. Set the number of input and output data

➢ Enter for the output module type and size of the output module. Up to 4 output modules can be set. With the output modules the number of data is configured which the netTAP or netBRICK device should receive from the PROFIBUS-DP Master.

➢ Enter for the input module type and size of the input module. Up to 4 input modules can be set. With the input modules the number of data is configured which the netTAP or netBRICK device should send to the PROFIBUS-DP Master.

Note: The input and output modules are ‚with consistence'.

© Hilscher, 2009-2010

5. Set more parameter

➢ If necessary, set more parameter. The parameter are described in the following section *PROFIBUS-DP Slave Parameter* on page 108.

Note: Default values can be used normally for the other parameters. Depending on additionally requirements and the design of the PROFIBUS network as well as the used devices parameters needs to be adjusted if necessary.

6. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

↬ The signal configuration dialog opens

7. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

8. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

↬ The configuration dialog closes

© Hilscher, 2009-2010

4.21.1 PROFIBUS-DP Slave Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Automatic communication start	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Off
I/O Data Status	Status of the input or the output data. For each input and output data the following status information (in Byte) is memorized in the dual-port memory: Status 0 = None (default) Status 1 = 1 Byte (for future use) Status 2 = 4 Byte (for future use)	Default: None
Ident		
Ident Number	PROFIBUS Identification Number If ‘Enabled’ is unchecked, the default value is used.	0x00000000 ... 0x0000FFFF (hex), Default: NT 50: 0x000000C99 (hex), NT 100: 0x000000C0E (hex), NB 100: 0x00000C9C (hex)
Bus		
Station Address	PROFIBUS address of the device	0 … 126
Baudrate	Network Baud Rate	9,6 kBit/s 19,2 kBit/s 93,75 kBit/s 187,5 kBit/s 500 kBit/s 1,5 MBit/s 3 MBit/s 6 MBit/s 12 MBit/s 31,25 kBit/s 45,45 kBit/s Auto-Detect Default NT 50, NT 100, NB 100: Auto-Detect
Extras	Sync supported: If checked, the Slave stack supports the SYNC command or the SYNC mode is activated.	Default NT 50, NT 100, NB 100: checked
	DPV1 Enable: If checked, DPV1 is supported or the DPV1 functions are activated.	
	Freeze supported: If checked, the Slave stack supports the FREEZE command or the FREEZE mode is activated.	
	Address change not allowed: If checked, the Slave stack supports the Set Slave Address command. The bus address can be changed via the Master.	
	Fail safe supported: If checked, the FAILSAFE operation is supported or the FAILSAFE mode is activated.	

Table 37: Parameters - PROFIBUS-DP Slave (Part 1)

© Hilscher, 2009-2010

Data		
Output or Input	**Module:** for output modules for input modules	 1 … 4 5 … 8
	Type: Byte or Word	„Byte“ (Default), „Word“ each with consistency
	Size: The number of Bytes or Words in the module.	0, 1, 2, 3, 4, 8, 12, 16, 20, 32, 64 (Byte, Words)
Output Data Bytes	Total of the output bytes of the modules 1 to 4	0 … 244, Default: NT 50, NT 100, NB 100: 2
Input Data Bytes	Total of the input bytes of the modules 5 to 8	0 … 244, Default: NT 50, NT 100, NB 100: 2
Custom data	Custom data: If unchecked the field Configuration Data shows the output and input identifier bytes, which results from the settings of the output modules and input modules. If checked the field Configuration Data is editable. Output and input identifier bytes can be entered into the Configuration Data field to configure the device. Then, the settings of the output modules and input modules have no meaning.	Default NT 50, NT 100, NB 100: unchecked
Configuration Data	Configuration data for the output and input identifier bytes. The identifier bytes consists of the **Type** and the **Size**. The identifier bytes are the general identifier bytes according to the PROFIBUS standard. Depending on the setting of the field Custom data, the field Configuration data is for display only or an entry field.	Default: A1, 91 hex

Table 38: Parameters - PROFIBUS-DP Slave (Part 2)

Note: The input and output modules each work with ’consistency’.

© Hilscher, 2009-2010

4.21.2 Settings at the used PROFIBUS-DP Master

Device Description File

To configure the Master a GSD file (device description file) is required. The GSD file is on the delivered DVD in the directory EDS/PROFIBUS:

Device	GSD File
NT 50-XX-DP or NT 50-DP-XX	HIL_0C99.GSD
NT 100-XX-DP or NT 100-DP-XX	HIL_0C0E.GSD
NB 100-XX-DP or NB 100-DP-XX	HIL_0C9C.GSD

Table 39: GSD File Names for netTAP and netBRICK

Configuration

Note: The settings in the used Master must comply with the settings in the Slave, to establish a communication. Important parameters are: Station Address, Ident Number, Baudrate and Config Data (the configuration data for the output and input length).

In order that the PROFIBUS-DP Master can communicate with the netTAP or netBRICK device as PROFIBUS-DP Slave:

- Enter at the PROFIBUS-DP Master the station address, which is configured in the netTAP or netBRICK device.
- Use for the configuration of the PROFIBUS-DP Master module(s) e. g. ‚64 Bytes Out', for sending data to the netTAP or netBRICK device. Use the module(s), which are configured in the netTAP or netBRICK device. (Modules with the size of zero in the netTAP or netBRICK configuration needs not to be considered for the configuration of the PROFIBUS-DP Master.)
- Use for the configuration of the PROFIBUS-DP Master module(s) e. g. ‚32 Bytes In', for receiving data from the netTAP or netBRICK device. Use the module(s), which are configured in the netTAP or netBRICK device. (Modules with the size of zero in the netTAP or netBRICK configuration needs not to be considered for the configuration of the PROFIBUS-DP Master.)

© Hilscher, 2009-2010

4.22 Configuration Modbus RTU Master/Slave

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as Modbus RTU Master or Slave needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > Modbus RTU**

↬ The Modbus RTU settings dialog opens.

↬ In the navigation area the entry **Configuration** under the folder **Configuration** is selected and the settings dialog is displayed.

2. Set protocol mode

➢ Set the **Protocol Mode** to "**I/O Slave**", if the netTAP or netBRICK device should work as Modbus RTU Slave

➢ Set for the **Protocol Mode** to "**Master**", if the netTAP or netBRICK device should work as Modbus RTU Master

↬ If you have set the **Protocol Mode** to "**Master**", then the entry **Command Table** appears in the navigation are

3. Set Modbus RTU Parameter

➢ Set or select in the configuration dialog the Modbus Address, the interface type, RTS control, baudrate, number of stop bits and parity for the netTAP or netBRICK device. The settings are described in the following section *Modbus RTU Parameter* on page 113.

➢ If you have set the **Protocol Mode** to "**Master**", then set the response timeout and the send retries.

4. Command Table

➢ If you have set the **Protocol Mode** to "**Master**", then select in the navigation area the entry **Command Table**. Specify in the Command Table which data the Modbus RTU Master has to transfer to and from the Modbus RTU Slave by reading or writing. The Command Table is described in detail in section *Modbus RTU Master* on page 114.

Note: To know, which data address with which data of the Modbus RTU Slave devices is provided for reading or for writing, refer to the device description of the Modbus RTU Slave device manufacturer.

© Hilscher, 2009-2010

5. Open the signal configuration dialog

➢ Select in the navigation area under the folder **Configuration** the entry **Signal Configuration**.

➧ The signal configuration dialog opens

6. Signal configuration

➢ If you want to assign own signal names, then enter your signal names. Further information is described in section *Signal Configuration* on page 140.

7. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.22.1 Modbus RTU Parameter

Parameter	Meaning	Range of Value/Value
Interface		
Bus Startup	Communication startsautomatically	Automatic (Default)
Watchdog Time [ms]	Watchdog time within which the device watchdog must be retriggered from the application program while the application program monitoring is activated. When the watchdog time value is equal to 0 the application program monitoring is deactivated.	[0, 20 … 65535] ms, default = 1000 ms, 0 = Watchdog timer off
Address Mapping	Valid address range of the data.	0 ... 65535 (Default)
Data Swapping	Data-storage mode: No: Data will not be swapped Yes: Data will be swapped.	Data will be swapped (default), Data will not be swapped
Ident		
Device	Device code for netIC (currently no other option available for choice) If ‘Enabled’ is unchecked, the default value is used.	NT50, NT100 (Default), NB100 String
Bus		
Protocol Mode	Determines the Mode of operation: Master or Slave	Master, IO Slave (Default)
Response Timeout	Timeout value (specified in milliseconds) Only valid for Master mode: Defines the maximum waiting time of the master for an answer from the slave	10 … 10000 ms, Default: 1000
Modbus Address	Defines the own Modbus address of the Gateway device.	1 … 247, Default: 2
Send Retries	Defines number of telegram retries when connection fails. Relevant only for operation as master.	0 … 10 Default: 3
Interface type	Serial interface to be used	RS232, RS485, RS422, Default: RS232
RTS Control	Signal “Rquest to Send can be switched off or on. See hardware description	RTS Control Off (aus, default), RTS Control On (ein)
Baudrate	Modbus RTU Network Baud Rate. Defines the transmission rate.	1,2 kBit/s 2,4 kBit/s 4,8 kBit/s 9,6 kBit/s (default) 19,2 kBit/s 38,4 kBit/s 57,6 kBit/s 115,2 kBit/s
Stop Bits	Stop Bits: Defines the quantity of stop bits, 1 or 2, to be used in protocol for serial data communication	One stop bit (default) Two stop bits
Parity	Defines the parity bit for serial data communication.	None Even (default) Odd
Data		
Register	Number of register Not editable in protocol mode master	0 ... 2880, Default: 2880
Coils	Number of coils not editable	Default: Automatic

Table 40: Modbus RTU Parameters

© Hilscher, 2009-2010

4.22.2 Modbus RTU Master

4.22.2.1 For what the Command Table does serve?

For the Modbus RTU Master the **Command Table** is a list with commands for the reading or writing of data.

The **Command Table** is only relevant, if a device works as Master on the Modbus RTU.

From every command line the Modbus RTU Master produces a telegram for reading or for the writing of data to or from a Modbus RTU Slave device. For every command are indicated:

- the Modbus RTU Slave device address (Device Address),
- the Function code (Function Code),
- the data address in the Modbus RTU Slave device (Address),
- the number of data (Number of Registers/Coils),
- and the data address in the Modbus RTU Master device (Dual-Port Memory Address).

For writing telegrams (FC 5, 6, 15 and 16) in the column

- Trigger

you can decide, whether the writing telegrams are executed every time (Cyclic) or only at data modification (Change data).

Furthermore

- a Cycle time (Cycle Time)

can be set for every command.

The **Command Table** is processed from the first to the last entry (from above to below). After the execution of the last command the execution of the first command is started again.

For reading commands the Modbus RTU Master reads out data from the Modbus RTU Slave and saves them into its data memory.

For writing commands the Modbus RTU Master reads out data from its data memory and writes them into the Modbus RTU Slave.

The number of commands which can be defined depends from the firmware and from the Dual-Port Memory layout.

© Hilscher, 2009-2010

4.22.2.2 Parameter of the Command Table

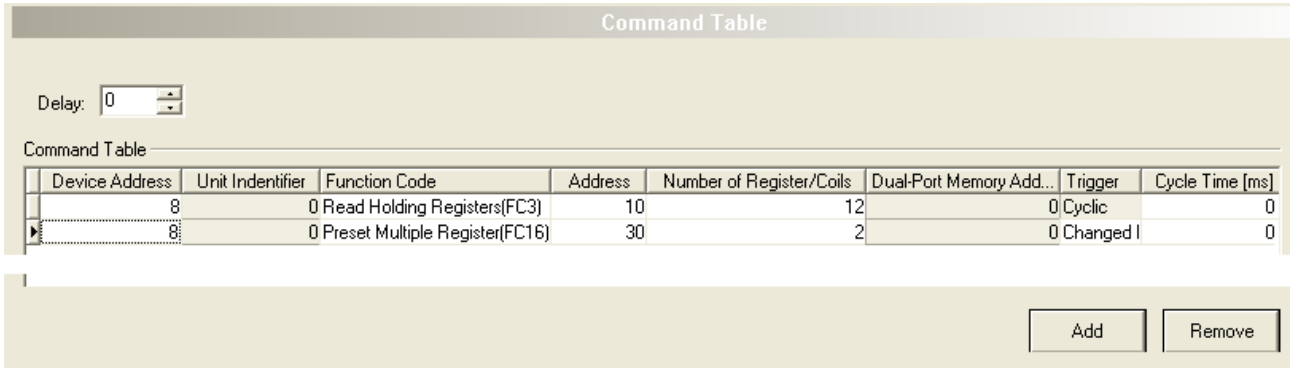


Figure 22: Pane Command Table

Each command contains the following parameters:

Parameter	Description
Device Address	Modbus RTU Slave device address
Unit Identifier	(not relevant here)
Function Code	Function code
Address	data address in the Modbus RTU Slave device
Number of Registers/Coils	Number of data
Dual-Port Memory Address	Data address in the Modbus RTU Master device (= Byte address in the process image of the Modbus RTU Master device)
Trigger	Trigger for writing commands
Cycle Time	Cycle time

Table 41: Parameter of the Command Table

© Hilscher, 2009-2010

<table>
<tr><th>Parameter</th><th>Meaning</th><th>Range of Value/Value</th></tr>
<tr><td>Device Address</td><td>Specifies the Modbus RTU Slave device address, from which the Modbus RTU master device reads out the data or into which it writes the data.</td><td>For Modbus RTU: 1 … 247
Default: 1</td></tr>
<tr><td>Unit Identifier</td><td>(not relevant here)</td><td>Default: 0</td></tr>
<tr><td>Function Code Address</td><td>Specifies the function code for read and write requests [3]:
Read Coils(FC1)
Read Inputs(FC2)
Read Holding Registers(FC3)
Read Input Registers(FC4)
Force Single Coil(FC5)
Preset Single Register(FC6)
Force Multiple Coils(FC15)
Preset Multiple Registers(FC16)</td><td>1 / 2 / 3 / 4 / 5 / 6 / 15 / 16
Default: „Read Coils (FC1)“</td></tr>
<tr><td>Address</td><td>Indicates the data address in the Modbus RTU Slave device.
The parameter Address contains the register address in the Modbus RTU Slave device. The register address is counted for every function code beginning with 0.
The following table shows the assignment of the Address to the data address in the Modbus RTU Slave by Function code:
<table>
<tr><th rowspan="2">Parameter Address</th><th colspan="4">Data Address Modbus RTU Slave</th></tr>
<tr><th>FC 1
FC 5
FC 15</th><th>FC 2</th><th>FC 4</th><th>FC 3
FC 6
FC 16</th></tr>
<tr><td>0</td><td>1</td><td>10001</td><td>30001</td><td>40001</td></tr>
<tr><td>1</td><td>2</td><td>10002</td><td>30002</td><td>40002</td></tr>
<tr><td>2</td><td>3</td><td>10003</td><td>30003</td><td>40003</td></tr>
<tr><td>3</td><td>4</td><td>10004</td><td>30004</td><td>40004</td></tr>
<tr><td>4</td><td>5</td><td>10005</td><td>30005</td><td>40005</td></tr>
<tr><td>…</td><td>…</td><td>…</td><td>…</td><td>…</td></tr>
</table>

To know, which data address with which data of the Modbus RTU Slave devices is provided for reading or for writing, refer to the device description of the Modbus RTU Slave device manufacturer.</td><td>0 … 65.535
Default: 0</td></tr>
<tr><td>Number of Registers/Coils</td><td>Indicates the number of reading or writing data as registers or coils.
The maximum value depends from the function code.</td><td>1 … max. value, Default: 1
max. value for
FC1 = 2000 (Coils),
FC2 = 2000 (Coils),
FC3 = 125 (Registers),
FC4 = 125 (Registers),
FC5 = 1 (Coils),
FC6 = 1 (Registers),
FC15 = 1968 (Coils),
FC16 = 123 (Registers)</td></tr>
</table>

For more see next page

© Hilscher, 2009-2010

Parameter	Meaning	Range of Value/Value
Dual-Port Memory Address	Byte address in the process image of the Modbus RTU Master device for input or output data of the Master The configuration software calculates the byte address in the process image of the Master for the input or for the output data automatically. Because of this definition, for function codes for reading (FC 1, 2, 3 and 4) the data are assigned consecutively in the process image for input data and for function codes for writing (FC 5, 6, 15 and 16) the data are assigned consecutively in the process image for output data.	0 … 5759 Default: 0
Trigger	Defines for all function codes for writing (FC 5, 6, 15 and 16); if the command is to be executed cyclically (Cyclic) or only if the data have changed (Change Data). For function codes for reading (FC 1, 2, 3 und 4) this parameter is not used.	0 = Cyclic 1 = Changed Data Default: „Cyclic“
Cycle Time	The cycle time defines, after how many milliseconds a writing or a reading command shall be reexecuted. The default value „0 ms“ means, that the command is executed as fast as possible. Otherwise the execution can be defined in steps of 10 ms. With the cycle time the temporal execution of the command can be influenced. If since the last execution of the command the set cycle time or more time has been passed, the command is reexecuted. If since the last execution of the command less time has been passed then the cycle time, the command is skipped. *If e. g. from a Modbus RTU Slave data are required only every 10 seconds, as these one change only slowly, then for the cycle time enter 10.000 ms.*	0, 10, 20, 30, … 60.000 ms Default: 0 ms
Delay	Between the single commands a delay time can be parameterized. This sometimes is necessary to avoid a too high load of the connected Slave devices by a continuous communication. **0**: The commands are processed without delay. **1 … 60.000 ms**: Delay time in ms, the Modbus RTU Master waits, before starting the next command.	0 … 60.000 ms Default: 0

Table 42: Parameter of the Command Table

© Hilscher, 2009-2010

4.22.2.3 Adding/removing Command

To add or to remove data sets for commands to the Command Table, proceed as follows:

Add command:

- In the table **Command Table** put the cursor in the line below which a new data set for a command shall be added.
- Click **Add**.

⇘ Under the clicked line a new command is added.

Remove command:

- In the table **Command Table** put the cursor in the line with the data set to be deleted.
- Click **Remove**.

⇘ The command of the clicked line is not shown any more.

4.22.2.4 Setting Parameters

Important: When entering the parameters, make sure that these address valid data registers. (*Refer to the device description of the Modbus RTU Slave device manufacturer.*)

To set the single parameters in the Command Table, proceed as follows:

1. **Device Address**, **Function Code**, **Address**, **Number of Registers/Coils**:

The fields for these parameters are editable.

- Enter the values for these parameters to the corresponding table cells.

2. **Trigger**:

- In **Function Code** field select an entry for a writing command (FC 5, 6, 15 or 16).
- In the column **Trigger** select the entry „Cyclic“ or „Change Data“.

3. **Cycle Time**:

- Put the cursor to the table cell and set the cycle time using the spin box in steps of 10 ms.

© Hilscher, 2009-2010

4.22.2.5 Examples Modbus RTU Reading or Writing Command

Device Address	Unit Indentifier	Function Code	Address	Number of Register/Coils	Dual-Port Memory Add...	Trigger	Cycle Time [ms]
8	0	Read Holding Registers(FC3)	10	12	0	Cyclic	0
8	0	Preset Multiple Register(FC16)	30	2	0	Changed I	0

Figure 23: Examples - Reading Command with FC 3, Writing Command with FC16

Example Reading Command with FC 3: From the Modbus RTU Slave device with the Modbus address 8 from data address 40011 onwards 12 registers are read by function code 3. The data are assigned to the dual-port memory address 0.

Example Writing Command with FC 16: To the Modbus RTU Slave with the Modbus address 8 from data address 40031 onwards 12 registers are written by function code 16. The data are read from the process image of the Master from the dual-port memory address 0 onwards. The writing process is only performed, if the data in the process image of the Master on the dual-port memory addresses 0 to 3 have changed, as the parameter trigger is set to 'changed'.

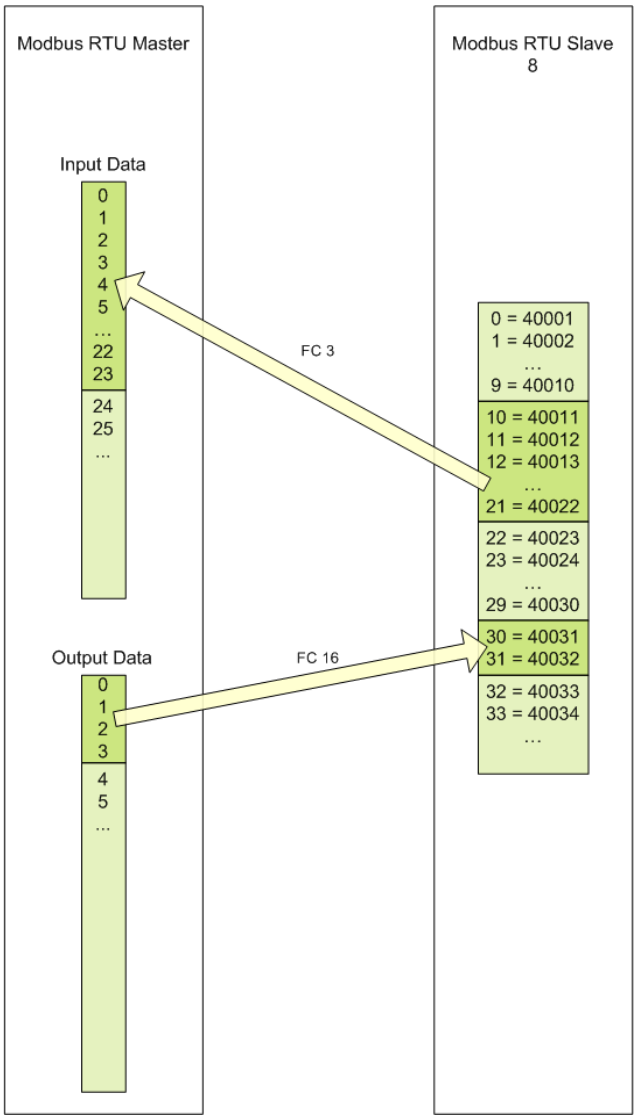


Figure 24: Examples - Reading Command with FC 3, Writing Command with FC16

© Hilscher, 2009-2010

4.22.2.6 Settings for Modbus RTU Slave

Note: The settings in the used Modbus RTU Slave(s) must comply with the settings in the netTAP or netBRICK device as Modbus RTU Master to establish a communication. Important parameters are: Modbus RTU address, baud rate, number of stop bits and parity setting.

© Hilscher, 2009-2010

4.22.3 Modbus RTU Slave

4.22.3.1 Settings at the used Modbus RTU Master

Note: The settings in the used Modbus RTU Master must comply with the settings in the netTAP or netBRICK device as Modbus RTU Slave to establish a communication. Important parameters are: Modbus RTU address, baud rate, number of stop bits and parity setting.

The Modbus RTU Master can access to the netTAP or netBRICK device as a Modbus RTU Slave using function codes 1, 2, 3, 4, 5, 6, 15 and 16

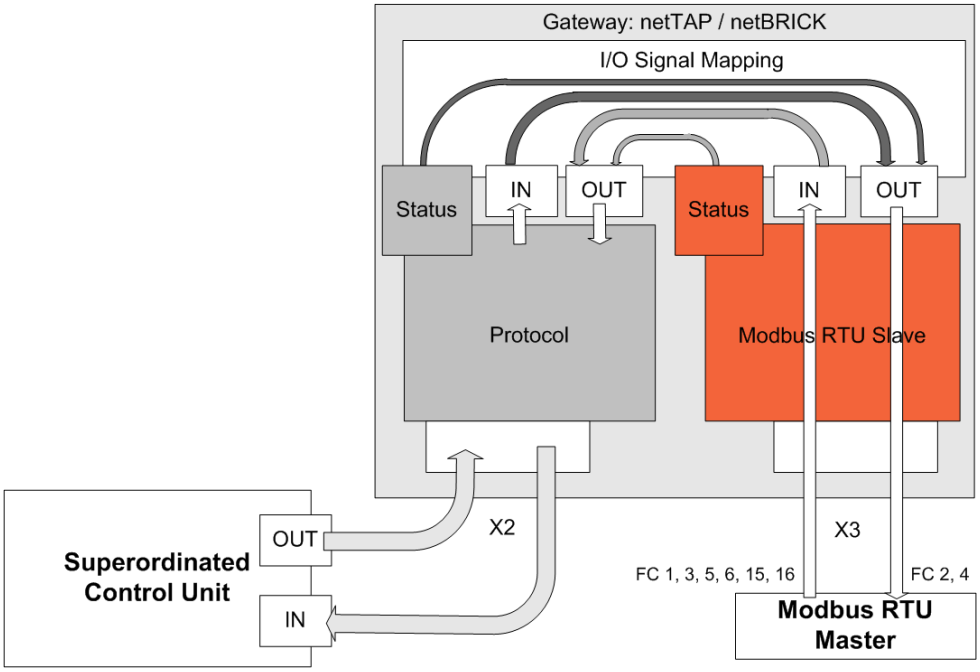


Figure 25: Access of the Modbus RTU Master to the netTAP and netBRICK Device (1)

The following table shows the register and coil addresses for many typical Modbus RTU Master systems:

Register or Coil of netTAP or netBRICK (Mapping)	Coil FC 1, FC 5, FC 15	Register FC 3, FC 6, FC 16
IN 0	1 … 16	40001
IN 1	17 … 32	40002
IN 2	33 … 48	40003
…	…	…

Register or Coil of netTAP or netBRICK (Mapping)	Coil FC 2	Register FC 4
OUT 0	10001 … 10016	30001
OUT 1	10017 … 10032	30002
OUT 2	10033 … 10048	30003
…	…	…

Figure 26: Access of the Modbus RTU Master to the netTAP and netBRICK Device (2)

Note: Typical Modbus RTU Master systems address on application level the first register with 40001 for function code 3 and 16. However, some Modbus RTU Master systems address on application level the first register with 0, others address the first register with 1. Read the documentation of the used Modbus RTU Master system to find out the used addressing of this system.

© Hilscher, 2009-2010

4.22.4 Modbus References

[1] MODBUS Application Protocol Specification V1.1, http://www.modbus.org/, 12/06/02

[2] MODBUS Messaging on TCP/IP Implementation Guide V1.0b, October 24, 2006

[3] MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1a, June 4, 2004, http://www.Modbus-IDA.org

© Hilscher, 2009-2010

4.23 Configuration ASCII

ASCII (= American Standard Code for Information Interchange) is a coding for characters. ASCII is denoted a communication for serial data exchange here and is usually a human readable code and

This type of communication contains however less definitions for the detailed communication procedure. Many manufacturer specific implementations exist.

Basic functions are send and receive of exactly one specified telegram.

The functionality depicted here for send and receive of telegrams can be used when one of the following condition is fulfilled:

- The send telegram has a fixed structure
- The receive telegram has a fixed structure
- The polling telegram (send telegram without data) has a fixed structure and the receive telegram has a fixed structure
- The polling telegram (send telegram with data) has a fixed structure and the receive telegram has a fixed structure

The structure of the serial send and receive telegram is specified by configuration. The part of the telegram marked as user data is transferred unchanged with a superordinated memory. For this data exchange a handshake mechanism has to be used.

ASCII should not be used, when

- different telegram structures for send telegrams or
- different telegram structures for receive telegrams or
- communication procedures with more complexity are to be used or
- the user data needs to be evaluated, processed respectively converted

Then use 'Serial with netSCRIPT'. With 'Serial with netSCRIPT' it can be reached by programming (script)

- that user data is evaluated, processed respectively converted,
- more complex communication procedures and,
- different telegram structures

can be realized.

© Hilscher, 2009-2010

The netTAP NT 50 and NT 100 device respectively the netBRICK NB 100 device as ASCII needs parameter.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > ASCII**

➧ The ASCII settings dialog opens.

➧ In the navigation area the entry **Settings** under the folder **Configuration** is selected and the settings dialog is displayed.

2. ASCII settings

➢ Select in the dialog window the interface type, RTS control, baudrate, number of data bits, number of stop bits and parity. The settings are described in the following section *ASCII Settings* on page 125.

3. Open ASCII parameter dialog

➢ Select in the navigation area under **Configuration** the entry **ASCII Paramter**.

➧ The dialog **ASCII Parameter** is displayed.

4. Set parameter

➢ Select the operating mode, set the telegram structure, set the timing and the size of the send and receive buffer. The parameter are described in section *ASCII Parameters* on page 125.

5. Signal configuration

➧ The signal configuration is done with default value automatically by SYCON.net software.

6. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

➧ The configuration dialog closes

© Hilscher, 2009-2010

4.23.1 ASCII Settings

Parameter	Meaning	Value Range/Value
Interface type	Type of the serial interface, which should be used	RS232 (Default), RS485, RS422
RTS Control	Specifies, if RTS control is activated or deactivated. When using the interface type RS-485, then "RTS Control on" has to be used.	RTS Control Off (Default), RTS Control On
Baud rate	Specifies the baud rate.	300 Bit/s 600 Bit/s 1200 Bit/s 2400 Bit/s 4800 Bit/s 9600 Bit/s (Default) 19200 Bit/s 38400 Bit/s 57600 Bit/s 115200 Bit/s
Data bits	Specifies the number of data bits, 7 or 8	7, 8 (Default)
Stop bits	Specifies the number of stop bits, 1 or 2	1 (Default) 2
Parity	Specifies the parity bit of the serial data communication. None means that no parity bit is used.	None (Default), Even, Odd

Table 43: ASCII Parameter

4.23.2 ASCII Parameters

The operating modes are:

- ‘Receive Only’ Mode
- ‘Send Only’ Mode
- Client Mode (first send, then receive)
- Server Mode (first receive, then send)

The further configuration parameters specify the

- the telegram structure of the send telegram,
- the telegram structure of the receive telegram,
- the timing,
- the size of the send and receive buffer

© Hilscher, 2009-2010

4.23.2.1 Telegram Structure in Data Stream

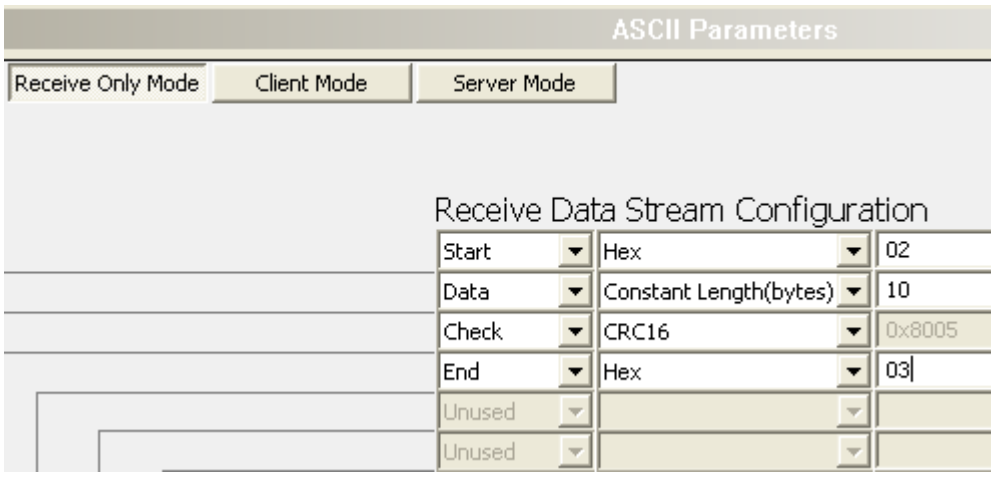
A send or receive telegram in the simplest case only consists only by user data. Many implementations however add to the user data further characters which have a certain meaning, such as a start character, an end character, a checksum or a device address.

A typical telegram structure is e. g.

Start | D a t a | Checksum | End

with for example (in hexadecimal notation):

[0x02] [0x38][0x33][0x33][0x37][0x38][0x30][0x33][0x37][0x36][0x33] [0x69][0xA5] [0x03]



The serial data stream of the send and receive telegram each can be defined with up to 10 structural elements.

© Hilscher, 2009-2010

Parameter	Description	Range of Value / Parameter Type
Unused	Structural element is not used	-
Start	Start character of the telegram consisting of one or more characters The information is given as an ASCII character (Char) or as hexadecimal (Hex). For the ASCII character 'STX' enter the hexadecimal value'02'. When sending, the start character is inserted in the telegram. When receiving, the start character is checked for equality.	Char Hex
Device	Device Address When sending, the device address is inserted in the telegram. Thereby the receiving device is specified. When receiving, the device address is checked for equality.	Decimal, Character, Hex
Object	Object Index or Start Address of the data in the device When sending, the object index / start address is inserted in the telegram. When receiving, the object index / start address is checked for equality.	Decimal, Character, Hex
Command	Command Identifier When sending, the Command Identifier is inserted in the telegram. When receiving, the Command Identifier is checked for equality.	Character, Hex
Data	Length specification for Data Filed or Termination Character **Length Specification:** Constant Length (bytes): The data field has a fixed length. The information is given as number of bytes. Byte Number Data: The length specification is given in one byte. When sending, this byte is inserted in the telegram directly before the data field. When receiving, the value of this byte from the telegram is used as the length of the directly following data field. Word Number Data: The length specification is given in one word. When sending, this word is inserted in the telegram directly before the data area. When receiving, the value of this word from the telegram is used as the length of the directly following data field. **Termination Character:** Character Terminated: The data field is terminated by a fixed end identifier consisting of one byte. When sending, this termination character is inserted in the telegram after the data field. When receiving, this termination character is used to detect the end of the data field. The termination character is not part of the data field.	Constant Length (Bytes) Byte Number Data Word Number Data Character Terminated
End	End of the telegram character consisting of one or more characters The information is given as an ASCII character (Char) or as hexadecimal (Hex). For the ASCII character 'ETX' enter the hexadecimal value'03'. When sending, the end character is inserted in the telegram. When receiving, the end character is checked for equality.	Char Hex

Table 44: Telegram Structure Part 1

© Hilscher, 2009-2010

Parameter	Description	Range of Value / Parameter Type
Check	Checksum CRC8: Any data of the telegram before the checksum are inserted in the initial value zero and the 1-byte result is used as checksum. CRC16: Any data of the telegram before the checksum are inserted in the initial value zero with the polynomial 8005 and the 2-byte result is used as checksum. CRC32: Any data of the telegram before the checksum are inserted in the initial value zero with the polynomial 04C11DB7 and the 4-byte result is used as checksum. Exor: Any data of the telegram before the checksum are set off against the initial value zero with exclusive-or and the 1-byte result is used as checksum. When sending, the computed checksum is inserted in the telegram. When receiving, the checksum is calculated based on of the received characters and is checked for equality with the received checksum.	CRC8, CRC16, CRC32, Exor
Don't care	Characters with no meaning Constant Length (bytes): Specifies the number of bytes, which have no meaning and which should be ignored. When sending, the number of characters with a value of zero is inserted in the telegram. When receiving, the number of characters is ignored and thus filtered out from the telegram.	Constant Length (Bytes)

Table 45: Telegram Structure Part 2

Type	Description
Hex	Hexadecimals The entry is made as a hexadecimal value. A hexadecimal value consists of 2 characters in the range 00 to FF and result in one byte. Each character has the range of value 0, 1, 2, ..., 9, A, B, ..., F.
Char	Character The entry is made as a character. A character is one of the following characters: A-Z, a-z, 0-9, !, ", §, $, %, &, /, (,), =, ?, , ; -, _, +, *
Decimal	Decimal value The entry is made as a decimal value.

Table 46: Parameter Types

Example: The specification of the character A is as hexadecimal value '41', as character 'A' or as a decimal value 65.

© Hilscher, 2009-2010

ASCII Character Table

ASCII Hex	ASCII Dez	Character	ASCII Hex	ASCII Dez	Character	ASCII Hex	ASCII Dez	Character	ASCII Hex	ASCII Dez	Character
00	0	NUL	20	32	SP	40	64	@	60	96	`
01	1	SOH ^A	21	33	!	41	65	A	61	97	a
02	2	STX ^B	22	34	"	42	66	B	62	98	b
03	3	ETX ^C	23	35	#	43	67	C	63	99	c
04	4	EOT ^D	24	36	$	44	68	D	64	100	d
05	5	ENQ ^E	25	37	%	45	69	E	65	101	e
06	6	ACK ^F	26	38	&	46	70	F	66	102	f
07	7	BEL ^G	27	39	'	47	71	G	67	103	g
08	8	BS ^H	28	40	(	48	72	H	68	104	h
09	9	TAB ^I	29	41	)	49	73	I	69	105	i
0A	10	LF ^J	2A	42	*	4A	74	J	6A	106	j
0B	11	VT ^K	2B	43	+	4B	75	K	6B	107	k
0C	12	FF ^L	2C	44	,	4C	76	L	6C	108	l
0D	13	CR ^M	2D	45	-	4D	77	M	6D	109	m
0E	14	SO ^N	2E	46	.	4E	78	N	6E	110	n
0F	15	SI ^O	2F	47	/	4F	79	O	6F	111	o
10	16	DLE ^P	30	48	0	50	80	P	70	112	p
11	17	DC1 ^Q	31	49	1	51	81	Q	71	113	q
12	18	DC2 ^R	32	50	2	52	82	R	72	114	r
13	19	DC3 ^S	33	51	3	53	83	S	73	115	s
14	20	DC4 ^T	34	52	4	54	84	T	74	116	t
15	21	NAK ^U	35	53	5	55	85	U	75	117	u
16	22	SYN ^V	36	54	6	56	86	V	76	118	v
17	23	ETB ^W	37	55	7	57	87	W	77	119	w
18	24	CAN ^X	38	56	8	58	88	X	78	120	x
19	25	EM ^Y	39	57	9	59	89	Y	79	121	y
1A	26	SUB ^Z	3A	58	:	5A	90	Z	7A	122	z
1B	27	Esc	3B	59	;	5B	91	[	7B	123	{
1C	28	FS	3C	60	<	5C	92	\	7C	124	|
1D	29	GS	3D	61	=	5D	93	]	7D	125	}
1E	30	RS	3E	62	>	5E	94	^	7E	126	~
1F	31	US	3F	63	?	5F	95	_	7F	127	DEL

Table 47: ASCII Character Table

© Hilscher, 2009-2010

4.23.2.2 Size of the Send and Receive Buffer

The size of the send or receive buffer is configured.

Note: The send telegram must fit into the send buffer. The receive telegram must fit into the receive buffer.

Parameter	Description	Range of Value
Sendbuffer Size	Size of the send buffer in bytes	0 … 1024 Default: 512 Bytes
Receivebuffer Size	Size of the receive buffer in bytes	0 … 1024 Default: 512 Bytes

Table 48: Size of the Send and Receive Buffer

4.23.2.3 Operating Modes and Timing

The modes are:

- ‘Receive Only’ Mode
- ‘Send Only’ Mode
- Client Mode (first send, then receive)
- Server Mode (first receive, then send)

© Hilscher, 2009-2010

4.23.2.4 'Send Only' Mode

In the 'Send Only' mode the device only sends. The remote device only receives.

The send operation can be triggered or performed cyclically from the network superordinated to ASCII.

- triggered

With each handshake of the superordinated control unit a telegram is send.

Setting the timing parameter **Send Cycle Time** to zero.

- Cyclic

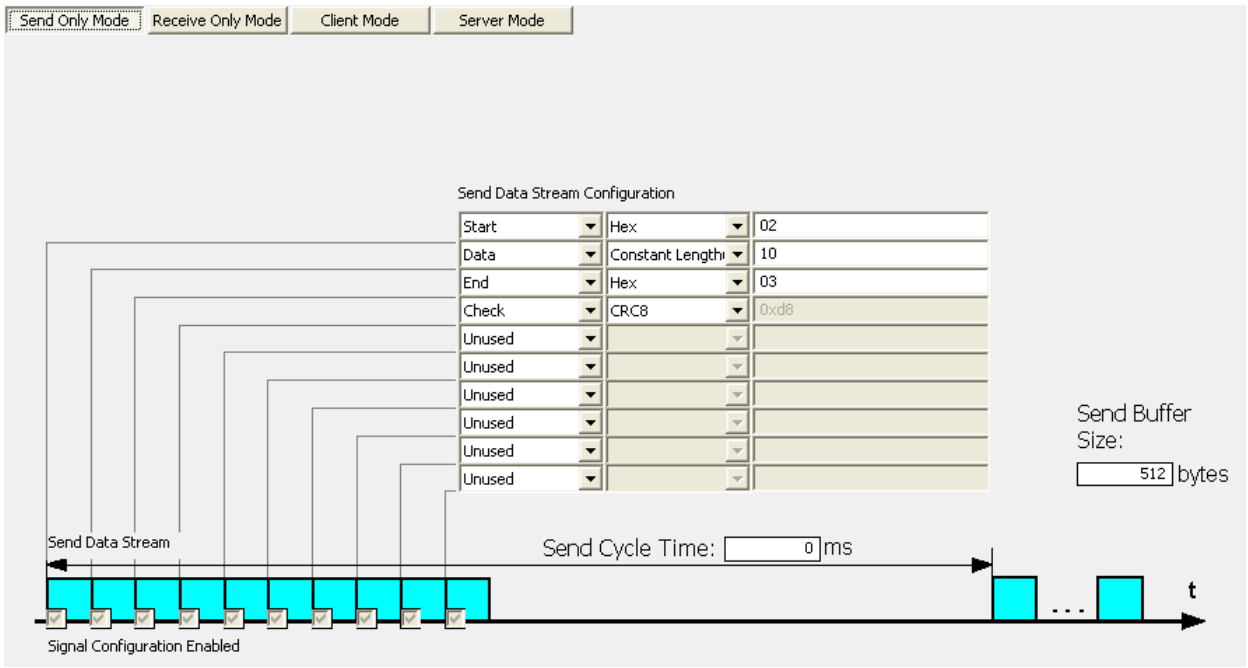
The device sends in a constant cycle.

With each handshake of the superordinated control unit the send data initially are updated in the internal buffer and sent during the next sending cycle.

Setting the timing parameter **Send Cycle Time** to the cycle time (unequal to zero).

Parameter	Description	Range of Value
Send Cycle Time	Specifies whether the telegram will be sent cyclically or triggered. The value (unequal to zero) specifies the cycle time of the send telegram. The value 0 specifies that the send telegram is triggered.	0 … 60000 Default: 0

Table 49: Timing Parameters for the Operating 'Send Only' Mode



© Hilscher, 2009-2010

4.23.2.5 'Receive Only' Mode

In the 'Receive Only' mode the device only receives. The remote device only sends.

When configuring the telegram structure the method is defined how the end of the telegram is detected:

• Character

• Length

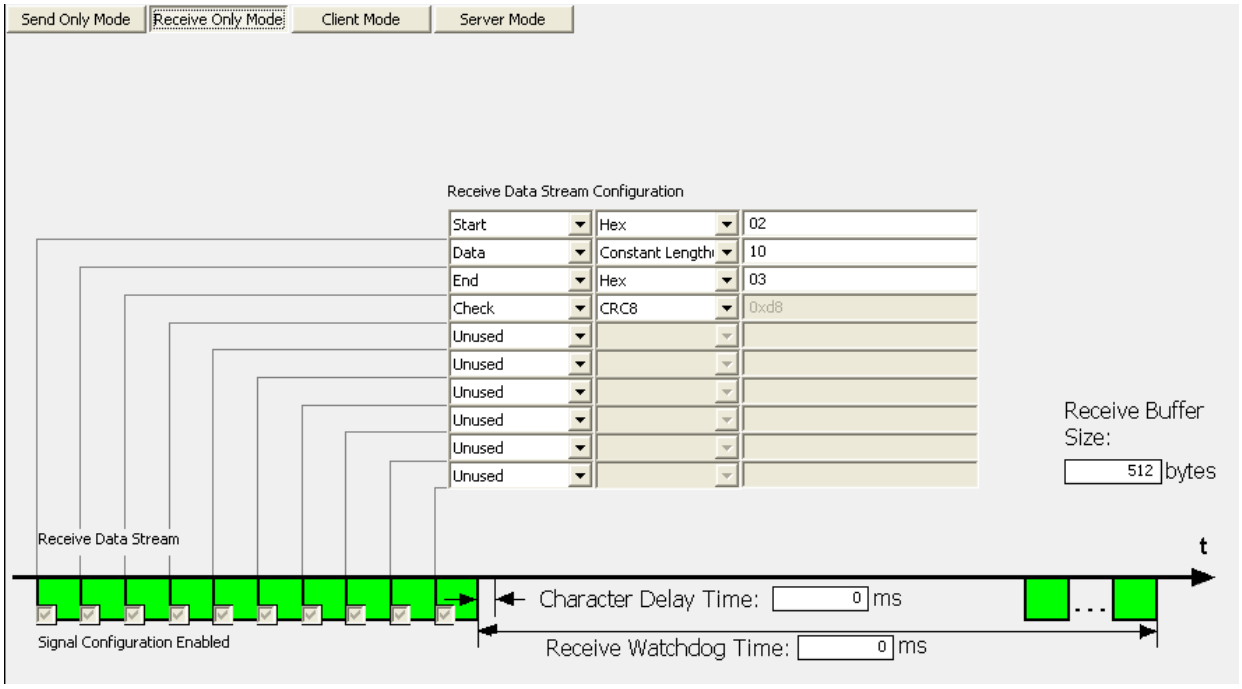
• Time

Each telegram received is indicated to the superordinated control unit by handshake. The superordinated control unit must acknowledge the reception. Only after the acknowledgement the reception of a further telegram can be indicated to the superordinated control unit.

Parameter	Description	Range of Value
Receive Watchdog Time	Specifies whether the reception of subsequent telegrams is monitored temporally. Thereby the remote device can be monitored. The time is taken from telegram end to telegram end. The value (unequal to zero) specifies in what time the remote device must have sent the next telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that the reception of subsequent telegrams is not monitored.	0 … 60000 Default: 0
Character Delay Time	Specifies whether the time between two characters during reception is monitored temporally. The value (unequal to zero) specifies in what time the remote device must have sent the next telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that no monitoring is performed.	0 … 2000 Default: 0

Table 50: Timing Parameters for the Operating 'Receive Only' Mode

© Hilscher, 2009-2010

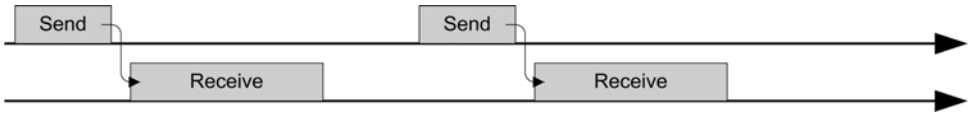


© Hilscher, 2009-2010

4.23.2.6 Client Mode (First Send, Then Receive)

In the 'Client Mode' the device sends a telegram to the remote device. The remote device then sends a telegram which is received by the device.

By this a polling (querying) of the remote devices can be performed. The device can send a polling telegram with or without user data.



The send operation can be triggered or performed cyclically.

• triggered

With each handshake of the superordinated control unit a telegram is send.

Setting the timing parameter **Send Cycle Time** to zero.

The remote device then sends a telegram which is received by the device. The device can monitor the beginning of the receive telegram temporally via the **Response Timeout** parameter.

The superordinated control unit must activate the send operation by handshake. The superordinated control unit must acknowledge each reception by handshake.

- Cyclic

The device sends in a constant cycle.

With each handshake of the host the send data initially are updated in the internal buffer and sent during the next sending cycle.

Setting the timing parameter **Send Cycle Time** to the cycle time (unequal to zero).

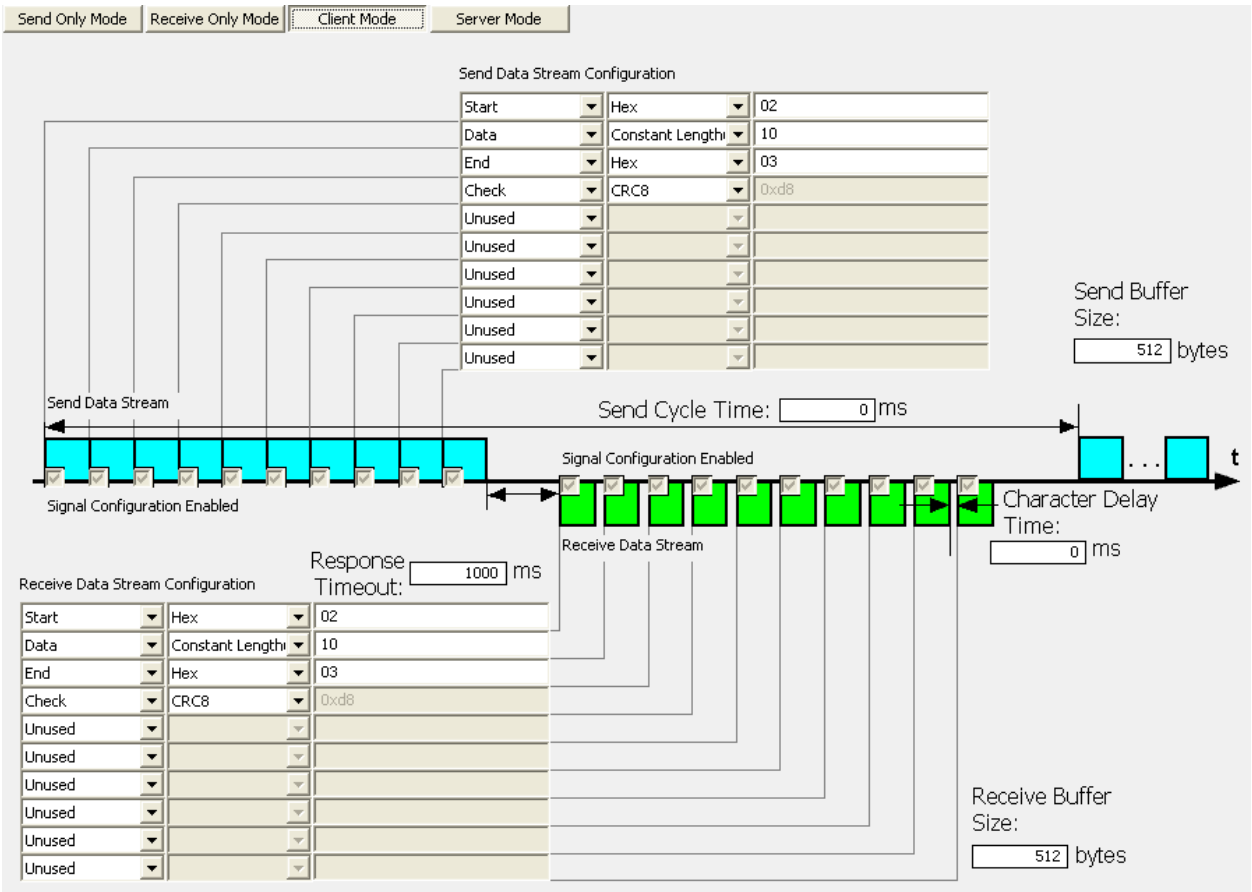
The remote device then sends a telegram which is received by the device. The device can monitor the beginning of the receive telegram temporally via the **Response Timeout** parameter.

The superordinated control unit must activate the send data by handshake. The superordinated control unit must acknowledge each reception by handshake.

© Hilscher, 2009-2010

Parameter	Description	Range of Value
Send Cycle Time	Specifies whether the telegram will be sent cyclically or triggered. The value (unequal to zero) specifies the cycle time of the send telegram. The value 0 specifies that the send telegram is triggered.	0 … 60000 Default: 0
Response Timeout	Specifies whether the reception of the response telegram is monitored temporally. Thereby the remote device can be monitored. The time is taken from telegram end to telegram end. The value (unequal to zero) specifies in what time the remote device must have sent the response telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that the reception of response telegrams is not monitored.	0 … 2000 Default: 0
Character Delay Time	Specifies whether the time between two characters during reception is monitored temporally. The value (unequal to zero) specifies in what time the remote device must have sent the next telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that no monitoring is performed.	0 … 2000 Default: 0

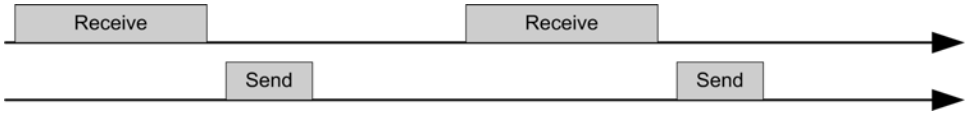
Table 51: Timing Parameters for the Operating Mode 'Client Mode'



© Hilscher, 2009-2010

4.23.2.7 Server Mode (First Receive, Then Send)

In the 'Server Mode' the unit receives a telegram from the remote device. The device then sends a telegram to the remote device.



When configuring the telegram structure the method is defined how the end of the telegram is detected:

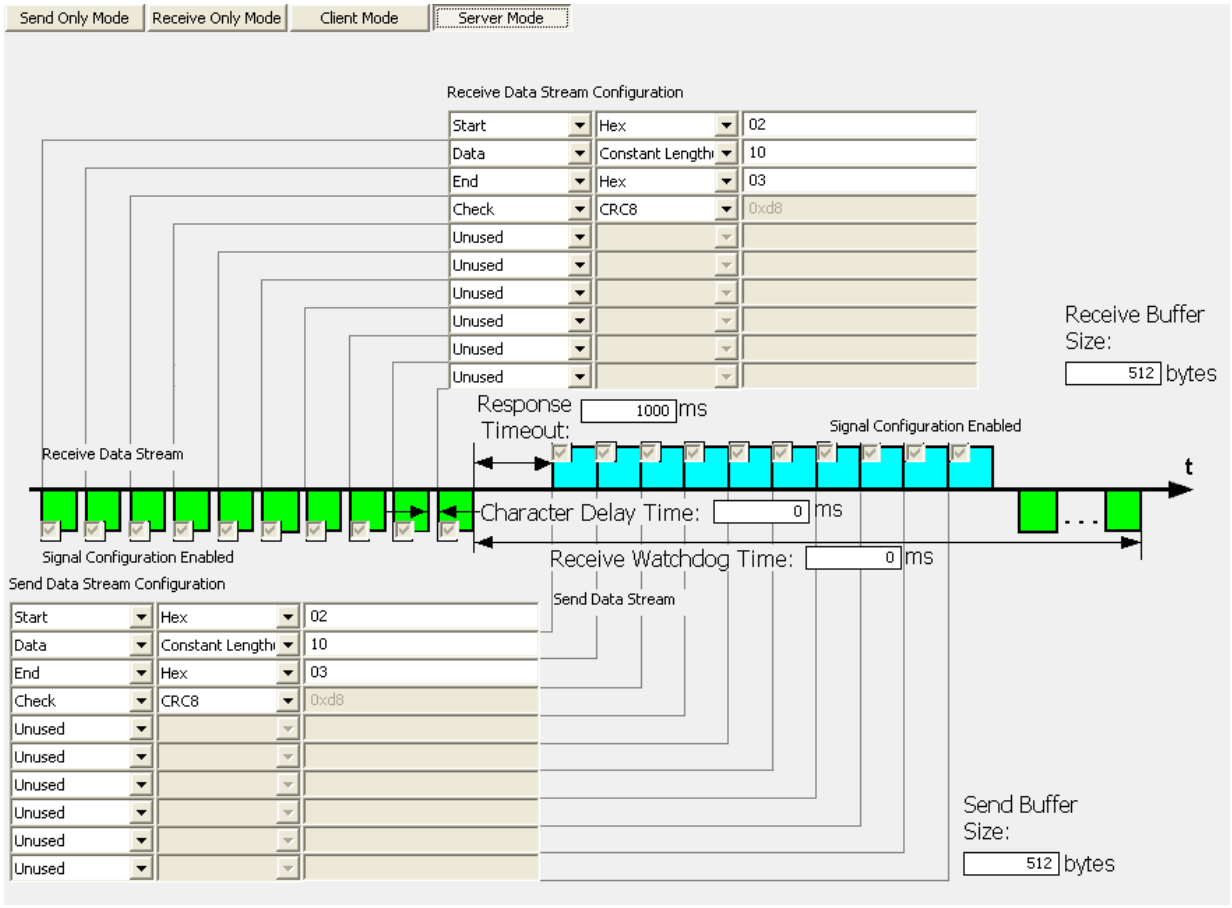
- Character
- Length
- Time

The superordinated control unit must acknowledge each reception by handshake. The superordinated control unit must activate by handshake the send operation.

Parameter	Description	Range of Value
Receive Watchdog Time	Specifies whether the reception of subsequent telegrams is monitored temporally. Thereby the remote device can be monitored. The time is taken from telegram end to telegram end. The value (unequal to zero) specifies in what time the remote device must have sent the next telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that the reception of subsequent telegrams is not monitored.	0 … 60000 Default: 0
Response Timeout	Specifies whether the reception of the response telegram is monitored temporally. Thereby the remote device can be monitored. The time is taken from telegram end to telegram end. The value (unequal to zero) specifies in what time the remote device must have sent the response telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that the reception of response telegrams is not monitored.	0 … 2000 Default: 0
Character Delay Time	Specifies whether the time between two characters during reception is monitored temporally. The value (unequal to zero) specifies in what time the remote device must have sent the next telegram. If the time is exceeded, then an error is reported to the host. The value 0 specifies that no monitoring is performed.	0 … 2000 Default: 0

Table 52: Timing Parameters for the Operating Mode 'Server Mode'

© Hilscher, 2009-2010



4.23.3 Settings for the ASCII Remote Device

Note: The settings in the used remote device must comply with the settings in the netTAP or netBRICK device to establish a communication. Important parameters are: Interface type, baud rate, data bits, parity check and respectively RTS control.

© Hilscher, 2009-2010

4.24 Configuration ‘Serial with netSCRIPT’

The netTAP NT 100 device respectively the netBRICK NB 100 device as ‘serial with netSCRIPT’ needs parameter and beyond a script has to be created and loaded.

These parameter can be edited as follows:

1. Open the configuration dialog

➢ Select from the context menu of the netTAP respectively the netBRICK symbol the entry **Configuration > netSCRIPT**

↬ The netSCRIPT settings dialog opens.

↬ In the navigation area the entry **Settings** under the folder **Configuration** is selected and the settings dialog is displayed.

2. netSCRIPT settings

➢ Select in the dialog window the interface type, RTS control, baudrate, number of data bits, number of stop bits and parity. The settings are described in the following section *netSCRIPT Settings* on page 139

3. Signal configuration

↬ The signal configuration is done with default value automatically by SYCON.net software.

4. Close configuration dialog

➢ Click on **OK** to save the parameter or click on **Cancel**, if the parameter should not be saved.

↬ The configuration dialog closes

Note: The programming of a script, loading the script into the netTAP or netBRICK device, the debugging of the script is described in detail in an own manual: netSCRIPT Programming Language for Serial Communication UM xx EN.pdf

© Hilscher, 2009-2010

4.24.1 netSCRIPT Settings

Parameter	Meaning	Value Range/Value
Interface type	Type of the serial interface, which should be used	RS232 (Default), RS485, RS422
RTS Control	Specifies, if RTS control is activated or deactivated. When using the interface type RS-485, then "RTS Control on" has to be used.	RTS Control Off (Default), RTS Control On
Baud rate	Specifies the baud rate.	300 Bit/s 600 Bit/s 1200 Bit/s 2400 Bit/s 4800 Bit/s 9600 Bit/s (Default) 19200 Bit/s 38400 Bit/s 57600 Bit/s 115200 Bit/s
Data bits	Specifies the number of data bits, 7 or 8	7, 8 (Default)
Stop bits	Specifies the number of stop bits, 1 or 2	1 (Default) 2
Parity	Specifies the parity bit of the serial data communication. None means that no parity bit is used.	None (Default), Even, Odd

Table 53: netSCRIPT Settings

4.24.2 Settings for the ASCII Remote Device

Note: The settings in the used remote device must comply with the settings in the netTAP or netBRICK device to establish a communication. Important parameters are: Interface type, baud rate, data bits, parity check and respectively RTS control.

© Hilscher, 2009-2010

4.25 Signal Configuration

At the pane **Signal Configuration** the data structure and the signal name of the in- and output signals is displayed. These signal are used for the signal mapping. Here for the single signals respectively the names of the signal names are set or the data types are configured.

SYCON.net generates for the signal names default names. These names can be overwritten by the user.

Parameter	Meaning	Range of Value/ Value
Type	Type of the input or output signals	String
Tag	Signal name for the single input or output signals. The name can be set by the user.	String
Data type	Data type of the single input or output signals. Depending by the used AS-Interface Slave profile the user can select the data type from a list.	BIT, WORD, SIGNED16, UNSIGNED16, etc., Default: depends on the used protocol
IO Type	Input or output signal	input, output

Table 54: Explanations to the Dialog Pane Signal Configuration

© Hilscher, 2009-2010

5 Diagnosis

5.1 Overview Diagnosis

The dialog **Diagnosis** serves to diagnose the device behavior and communication errors. For diagnosis the device must reside in online state.

The **Extended Diagnosis** helps to find communication and configuration errors, when default diagnosis fails.

Diagnosis Panes

The table below gives an overview for the individual **Diagnosis** dialog panes descriptions:

Section	Page
General Diagnosis	*142*
Firmware Diagnosis	*145*

Table 55: Descriptions of the Diagnosis Panes

Note: Accessing the **Diagnosis** panes of the netGateway DTM requires an online connection from the netGateway DTM to the netTAP NT 100 device.

For further information, refer to section *Connecting/Disconnecting Device* on page 146.

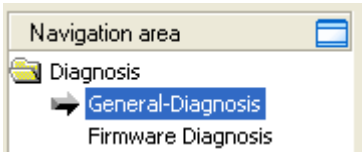


Figure 27: Navigation Area - Diagnosis

© Hilscher, 2009-2010

5.2 General Diagnosis

Information regarding the Device State and other general diagnosis parameters are displayed in the **General Diagnosis** dialog.

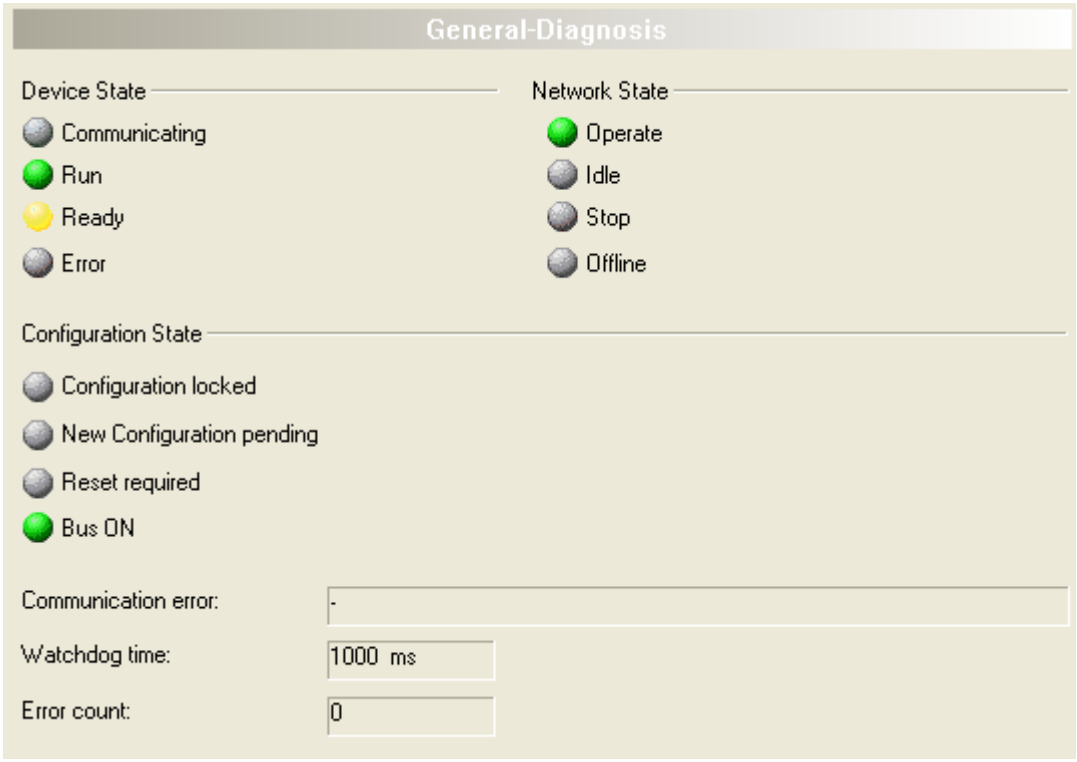


Figure 28: General Diagnosis

© Hilscher, 2009-2010

Indication	Meaning
Device State	
Communicating	Communicating: Shows that the netTAP firmware executes the network communication.
Run	Device is configured: Shows that the netTAP firmware has been configured correctly.
Ready	Ready: Shows that the netTAP firmware has been started correctly. The netTAP firmware waits for a configuration.
Error	Error: Shows that the netTAP firmware records a device status error. For further information to the error characteristics and the number of counted errors refer to the extended diagnosis.
Network State	
Operate	Operate: Shows that the netTAP firmware is in data exchange.
Idle	Idle: Shows that the netTAP firmware is in idle mode.
Stop	Stop: Shows that the netTAP firmware is in Stop state: There is no cyclic data exchange at the network. The netTAP firmware was stopped by the application program or it changed to the Stop state because of a bus error.
Offline	Offline: The netTAP firmware is offline pending it does not have a valid configuration.
Configuration State	
Configuration locked	Configuration locked: Shows that the netTAP firmware configuration is locked, to avoid the configuration data are typed over.
New Configura-tion pending	New Configuration pending: Shows that a new netTAP firmware configuration is available.
Reset required	Reset required: Shows that a firmware reset is required as a new netTAP firmware configuration has been loaded into the device.
Bus ON	Bus ON: Shows whether the bus communication was started or stopped. I. e., whether the device is active on the bus or no bus communication to the device is possible and no response telegrams are sent.

Table 56: Indication General Diagnosis

© Hilscher, 2009-2010

Parameter	Meaning
Communication Error	Communication Error: Shows the message text of the communication error. If the cause of the current error is resolved, „ – “ is displayed.
Watchdog time	Watchdog time: Shows the watchdog time in ms.
Error Count	Error Count: This field holds the total number of errors detected since power-up, respectively after reset. The protocol stack counts all sorts of errors in this field no matter if they were network related or caused internally.

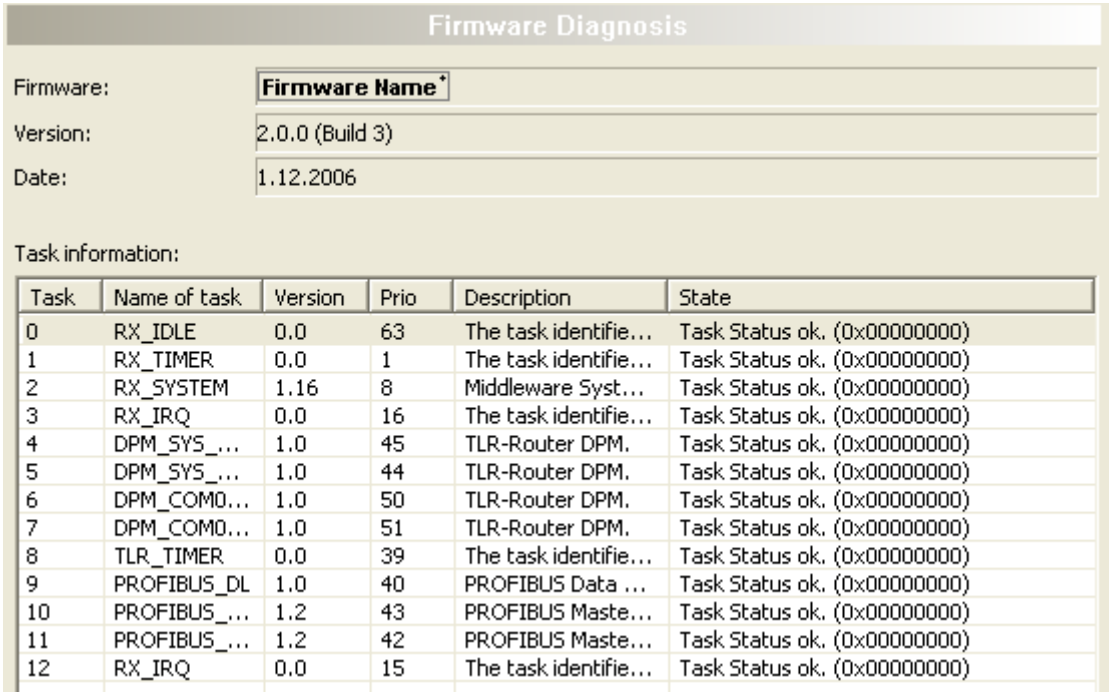
Table 57: Parameter General Diagnosis

© Hilscher, 2009-2010

5.3 Firmware Diagnosis

In the dialog **Firmware Diagnosis** the actual task information of the firmware is displayed.

Under **Firmware** or **Version** the name of the firmware and version (including the date) are indicated.



Task	Name of task	Version	Prio	Description	State
0	RX_IDLE	0.0	63	The task identifie...	Task Status ok. (0x00000000)
1	RX_TIMER	0.0	1	The task identifie...	Task Status ok. (0x00000000)
2	RX_SYSTEM	1.16	8	Middleware Syst...	Task Status ok. (0x00000000)
3	RX_IRQ	0.0	16	The task identifie...	Task Status ok. (0x00000000)
4	DPM_SYS_...	1.0	45	TLR-Router DPM.	Task Status ok. (0x00000000)
5	DPM_SYS_...	1.0	44	TLR-Router DPM.	Task Status ok. (0x00000000)
6	DPM_COM0...	1.0	50	TLR-Router DPM.	Task Status ok. (0x00000000)
7	DPM_COM0...	1.0	51	TLR-Router DPM.	Task Status ok. (0x00000000)
8	TLR_TIMER	0.0	39	The task identifie...	Task Status ok. (0x00000000)
9	PROFIBUS_DL	1.0	40	PROFIBUS Data ...	Task Status ok. (0x00000000)
10	PROFIBUS_...	1.2	43	PROFIBUS Maste...	Task Status ok. (0x00000000)
11	PROFIBUS_...	1.2	42	PROFIBUS Maste...	Task Status ok. (0x00000000)
12	RX_IRQ	0.0	15	The task identifie...	Task Status ok. (0x00000000)

*Figure 29: Firmware Diagnosis (*The name of the Firmware is displayed.)*

Task Information:

The table **Task Information** is listing the task information of the single firmware tasks.

Column	Meaning
Task	Task number
Task Name	Name of the task
Version	Version of the task
Prio	Priority of the task
Description	Description of the task
Status	Status of the task

Table 58: Description Table Task Information

© Hilscher, 2009-2010

6 Online Functions

6.1 Connecting/Disconnecting Device

Note: Several netGateway DTM functions e. g. **Diagnosis** or the configuration download in the FDT Framework require an online connection from the netGateway DTM to the netTAP NT 100 device.

Connecting Device

The following steps are needed to establish a connection from the netGateway DTM to a netTAP NT 100 device:

Under **Settings** in the **Driver** pane:

1. Select one or several driver/s.
2. Configure the driver, if necessary.

Under **Settings** in the **Device Assignment** pane:

3. Scan for the device.
4. Select the device and apply the selection.
5. In the DTM interface dialog select the **OK** button, to apply the selection and to close the DTM interface dialog.
6. Put a right-click on the netTAP symbol.
7. Select the **Connect** command from the right mouse button menu.

➮ The netTAP NT 100 device now is connected to the netGateway DTM via an online connection. In the network view the device description at the device symbol is displayed with a green colored background.

© Hilscher, 2009-2010

Disconnecting Device

To disconnect an online connection from the netTAP NT 100 device to a netGateway DTM take the following steps:

1. In the DTM interface dialog select the **OK** button, to close the DTM interface dialog.
2. Right-click on the netTAP symbol.
3. Select the **Disconnect** command from the context menu.

➪ Now the netTAP NT 100 device is disconnected from the DTM.

© Hilscher, 2009-2010

6.2 Download Configuration

Note: To download configuration parameter data to the netTAP NT 100 device an online connection from the netGateway DTM to the netTAP NT 100 device is required.

Further information can be found in the *Connecting/Disconnecting Device* section on page 146.

To transfer the configuration with the corresponding parameter data to the netTAP NT 100 device you must download the data to it using the frame application of the configuration software.

© Hilscher, 2009-2010

7 Error Numbers

7.1 Error Code Definition

For COM based application, like the ODM Server and ODM drivers, a common error definition is used similar to the Microsoft Windows HRESULT definition.

Error code definition:

COM Errors are HRESULTs, which are 32 bit values using the following layout:

```
 3 3 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1 1
 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0 9 8 7 6 5 4 3 2 1 0
+---+-+-+-----------------------+-------------------------------+
|Sev|C|R|     Facility          |               Code            |
+---+-+-+-----------------------+-------------------------------+

where
 Sev       - is the severity code:
        00 - Success
        01 - Informational
        10 - Warning
        11 - Error
 C         - is the Customer code flag
 R         - is a reserved bit
 Facility  - is the facility code
 Code      - is the facility's status code
```

In this common error definition, several error code regions are already reserved by Windows itself, the ODM and some other modules.

© Hilscher, 2009-2010

7.2 Overview Error Numbers

Module	Error Code Region
General Hardware Error Codes / RCX Operating System	*RCX Errors - MID_SYS Task Error Codes*: 0xC02B0001L to 0x8004BFFF to 0xC02B4B54L
	RCX Errors - Common Status Codes: 0x00000000L to 0xC000F006L
	RCX Errors - Status & Error Codes: 0x00000000L to 0xC0000008
ODM Server	*General ODM Error Codes*: 0x8004C700 to 0x8004C761
	General ODM Driver Error Codes: 0x8004C7A0 to 0x8004C7C2
ODM Drivers	*ODM cifX Driver Specific Errors*: 0x8004C001 to 0x8004C0A4
DBM	*ODM DBM V4 Error Codes*: 0xC004C810 to 0xC004C878

Table 59: Error Code Regions

The fieldbus specific error codes are described in the manuals of the corresponding protocol tasks.

© Hilscher, 2009-2010

7.3 General Hardware Error Codes

7.3.1 RCX Errors - MID_SYS Task Error Codes

ID of the Error Code	Value	Description
TLR_E_MID_SYS_QUE_ UNKNOWN	0xC02B0001L	Queue unknown.
TLR_E_MID_SYS_QUE_ IDX_UNKNOWN	0xC02B0002L	Queue table index does not exist.
TLR_E_MID_SYS_TSK_ UNKNOWN	0xC02B0003L	Task unknown.
TLR_E_MID_SYS_TSK_ IDX_UNKNOWN	0xC02B0004L	Task table index does not exist.
TLR_E_MID_SYS_TSK_ HANDLE_INVALID	0xC02B0005L	Task handle invalid.
TLR_E_MID_SYS_TSK_INFO_ IDX_UNKNOWN	0xC02B0006L	Task info field index unknown.
TLR_I_MID_SYS_FILE_ RETRANSMIT	0x402B0001L	The last data block was invalid, please retransmit.
TLR_E_MID_SYS_FILE_ XFR_TYPE_INVALID	0xC02B0007L	Requested transfer type invalid.
TLR_E_MID_SYS_FILE_ REQUEST_INCORRECT	0xC02B0008L	Request is incorrectly formatted i.e. wrong parameters.
TLR_E_MID_SYS_UNKNOWN_ PORT_INDEX	0xC02B0009L	Unknown port index.
TLR_E_MID_SYS_ROUTER_ TABLE_FULL	0xC02B000AL	Router Table is full.
TLR_E_MID_SYS_NO_SUCH_ ROUTER_IN_TABLE	0xC02B000BL	No such router in table.
TLR_E_MID_SYS_INSTANCE_ NOT_NULL	0xC02B000CL	Mid_Sys Instance is not 0.
TLR_E_MID_SYS_COMMAND_ INVALID	0xC02B000DL	Invalid command.
TLR_E_MID_SYS_TSK_ INVALID	0xC02B000EL	Invalid task handle.
TLR_E_MID_SYS_TSK_NOT_ A_USER_TASK	0xC02B000FL	Access denied. Not a user task (See Config-File).
TLR_E_MID_SYS_LOG_QUE_ NOT_SETTABLE	0xC02B0010L	Logical queue handle not settable.
TLR_E_MID_SYS_LOG_QUE_ NOT_INVALID	0xC02B0011L	Logical queue handle invalid.
TLR_E_MID_SYS_LOG_QUE_ NOT_SET	0xC02B0012L	Logical queue handle has not been set.
TLR_E_MID_SYS_LOG_QUE_ ALREADY_USED	0xC02B0013L	Logical queue handle is already in use.
TLR_E_MID_SYS_TSK_NO_ DEFAULT_QUEUE	0xC02B0014L	Task has no default process queue.
TLR_E_MID_SYS_MODULE_ INVALID	0xC02B0015L	Firmware Module is invalid. CRC-32 check failed.
TLR_E_MID_SYS_MODULE_ NOT_FOUND	0xC02B0016L	Firmware Module has not been found. Maybe it has not been downloaded before.
TLR_E_MID_SYS_MODULE_ RELOC_ERROR	0xC02B0017L	Firmware Module has an invalid reloc table.
TLR_E_MID_SYS_MODULE_ NO_INIT_TBL	0xC02B0018L	Firmware Module has no init table.
TLR_E_MID_SYS_MODULE_ NO_ENTRY_POINT	0xC02B0019L	Firmware Module has no code entry point.
TLR_E_MID_SYS_ACCESS_ DENIED_IN_LOCKED_STATE	0xC02B001AL	Access denied due to current operating conditions.
TLR_E_MID_SYS_INVALID_ FIRMWARE_SIZE	0xC02B001BL	Firmware does not fit into flash.
TLR_E_MID_SYS_MODULE_ RELOCATION_DISTANCE_ TOO_LONG	0xC02B001CL	The relocation distance is too long.

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
TLR_E_MID_SYS_SEC_ FAILED	0xC02B001DL	Access to the security flash failed.
TLR_E_MID_SYS_SEC_ DISABLED	0xC02B001EL	Security flash is disabled at firmware.
TLR_E_MID_SYS_FILE_ TRANSFER_IN_USE	0xC02B524CL	File Transfer in use.
TLR_E_MID_SYS_FILE_ TRANSFER_PACKET_ INVALID	0xC02B4444L	File Transfer Packet invalid.
TLR_E_MID_SYS_FILE_TRANSFER_ NOT_ACTIVE	0xC02B5342L	File Transfer is not active.
TLR_E_MID_SYS_FILE_TRANSFER_ INVALID	0xC02B5257L	File Transfer has invalid type code.
TLR_E_MID_SYS_FILE_CRC_ REPEATEDLY_WRONG	0xC02B4352L	File Transfer was tried repeatedly with a wrong CRC.
TLR_E_MID_SYS_FILE_TRANSFER_ TYPE_NOT_AVAILABLE	0xC02B4353L	Transfer Type is not available.
TLR_E_MID_SYS_PATH_INVALID	0xC02B5555L	File Path submitted in File Transfer was invalid.
TLR_E_MID_SYS_DRIVER_ CFG_TABLE_INIT_FUNCTION_ MISSING	0xC02BFFFFL	Driver Configuration Table Init Function missing.
TLR_E_MID_SYS_CONFIGURATION_ LOCKED	0xC02B4B54L	Configuration has been locked.

Table 60: RCX Errors - MID_SYS Task Error Codes (from Operating System)

© Hilscher, 2009-2010

7.3.2 RCX Errors - Common Status Codes

ID of the Error Code	Value	Description
TLR_S_OK	0x00000000L	Operation succeeded.
TLR_E_FAIL	0xC0000001L	Common error, detailed error information optionally present in the data area of packet.
TLR_E_UNEXPECTED	0xC0000002L	Unexpected failure.
TLR_E_OUTOFMEMORY	0xC0000003L	Ran out of memory.
TLR_E_UNKNOWN_COMMAND	0xC0000004L	Unknown Command in Packet received.
TLR_E_UNKNOWN_DESTINATION	0xC0000005L	Unknown Destination in Packet received.
TLR_E_UNKNOWN_DESTINATION_ID	0xC0000006L	Unknown Destination Id in Packet received.
TLR_E_INVALID_PACKET_LEN	0xC0000007L	Packet length is invalid.
TLR_E_INVALID_EXTENSION	0xC0000008L	Invalid Extension in Packet received.
RCX_COMM_WATCHDOG_TIMEOUT	0xC000000CL	Watchdog error occurred.
RCX_COMM_INIT_FAULT	0xC0000100L	General initialization fault.
RCX_COMM_DATABASE_ACCESS_ FAILED	0xC0000101L	Database access failture.
RCX_COMM_CONFIGURATION_ FAULT	0xC0000120L	General configuration fault.
RCX_COMM_INCONSISTENT_ DATA_SET	0xC0000121L	Inconsistent configuration data.
RCX_COMM_DATA_SET_MISMATCH	0xC0000122L	Configuration data set mismatch.
RCX_COMM_INSUFFICIENT_ LICENSE	0xC0000123L	Insufficient license.
RCX_COMM_PARAMETER_ERROR	0xC0000124L	Parameter error.
RCX_COMM_INVALID_NETWORK_ ADDRESS	0xC0000125L	Network address invalid.
RCX_COMM_NETWORK_FAULT	0xC0000140L	General communication fault.
RCX_COMM_CONNECTION_ CLOSED	0xC0000141L	Connection closed.
RCX_COMM_CONNECTION_ TIMEOUT	0xC0000142L	Connection timeout.
RCX_COMM_LONELY_NETWORK	0xC0000143L	Lonely network.
RCX_COMM_DUPLICATE_ NODE	0xC0000144L	Duplicate network address.
RCX_COMM_CABLE_ DISCONNECT	0xC0000145L	Cable disconnected.
TLR_S_FRAGMENTED	0x0000F005L	Fragment accepted.
TLR_E_RESET_REQUIRED	0xC000F006L	Reset required.

Table 61: RCX Errors - Common Status Codes (from Operating System)

© Hilscher, 2009-2010

7.3.3 RCX Errors - Status & Error Codes

ID of the Error Code	Value	Description
RCX_S_OK	0x00000000	SUCCESS, STATUS OKAY
RCX_S_QUE_UNKNOWN	0xC02B0001	UNKNOWN QUEUE
RCX_S_QUE_INDEX_UNKNOWN	0xC02B0002	UNKNOWN QUEUE INDEX
RCX_S_TASK_UNKNOWN	0xC02B0003	UNKNOWN TASK
RCX_S_TASK_INDEX_UNKNOWN	0xC02B0004	UNKNOWN TASK INDEX
RCX_S_TASK_HANDLE_INVALID	0xC02B0005	INVALID TASK HANDLE
RCX_S_TASK_INFO_IDX_UNKNOWN	0xC02B0006	UNKNOWN INDEX
RCX_S_FILE_XFR_TYPE_INVALID	0xC02B0007	INVALID TRANSFER TYPE
RCX_S_FILE_REQUEST_INCORRECT	0xC02B0008	INVALID FILE REQUEST
RCX_S_UNKNOWN_DESTINATION	0xC0000005	UNKNOWN DESTINATION
RCX_S_UNKNOWN_DESTINATION_ID	0xC0000006	UNKNOWN DESTINATION ID
RCX_S_INVALID_LENGTH	0xC0000007	INVALID LENGTH
RCX_S_UNKNOWN_COMMAND	0xC0000004	UNKNOWN COMMAND
RCX_S_INVALID_EXTENSION	0xC0000008	INVALID EXTENSION

Table 62: RCX Errors - Status & Error Codes (from Operating System)

7.3.3.1 RCX Errors - Status & Error Codes – Slave State

ID of the Error Code	Value	Description
RCX_SLAVE_STATE_UNDEFINED	0x00000000	UNDEFINED
RCX_SLAVE_STATE_OK	0x00000001	OK
RCX_SLAVE_STATE_FAILED	0x00000002	FAILED (at least one slave)

Table 63: RCX Errors - Status & Error Codes – Slave State (from Operating System)

© Hilscher, 2009-2010

7.4 ODM Error Codes

7.4.1 General ODM Error Codes

Symbol	Value	Description
CODM3_E_INTERNALERROR	0x8004C700	Internal ODM Error
ODM3_E_DESCRIPTION_ NOTFOUND	0x8004C701	Description not found in ODM database
CODM3_E_WRITEREGISTRY	0x8004C710	Error writing to the registry
CODM3_E_BAD_REGULAR_ EXPRESSION	0x8004C711	Invalid regular expression
CODM3_E_COMCATEGORIE_ MANAGER_FAILED	0x8004C712	Component Category Manager could not be instantiated
CODM3_E_COMCATEGORIE_ ENUMERATION_FAILED	0x8004C713	Driver could not be enumerated by the Category Manager
CODM3_E_CREATE_LOCAL_ BUFFER	0x8004C714	Error creating local buffers
CODM3_E_UNKNOWNHANDLE	0x8004C715	Unknown handle
CODM3_E_QUEUE_LIMIT_ REACHED	0x8004C717	Queue size limit for connection reached
CODM3_E_DATASIZE_ZERO	0x8004C718	Zero data length passed
CODM3_E_INVALID_DATA	0x8004C719	Invalid data content
CODM3_E_INVALID_MODE	0x8004C71A	Invalid mode
CODM3_E_DATABASE_READ	0x8004C71B	Error reading database
CODM3_E_CREATE_DEVICE_ THREAD	0x8004C750	Error creating device thread
CODM3_E_CREATE_DEVICE_ THREAD_STOP_EVENT	0x8004C751	Error creating device thread stop event
CODM3_E_CLIENT_NOT_ REGISTERED	0x8004C752	Client is not registered at the ODM
CODM3_E_NO_MORE_ CLIENTS	0x8004C753	Maximum number of clients reached
CODM3_E_MAX_CLIENT_ CONNECTIONS_REACHED	0x8004C754	Maximum number of client connections reached
CODM3_E_ENTRY_NOT_ FOUND	0x8004C755	Driver/device not found
CODM3_E_DRIVER_NOT_ FOUND	0x8004C757	The requested driver is unknown to the ODM
CODM3_E_DEVICE_ALREADY_ LOCKED	0x8004C758	Device is locked by another process
CODM3_E_DEVICE_ UNLOCKED_FAILED	0x8004C759	Device could not be unlocked, lock was set by another process
CODM3_E_DEVICE_ LOCK_NECCESSARY	0x8004C75A	Operation requires a device lock to be set
CODM3_E_DEVICE_ SUBSCRIPTIONLIMIT	0x8004C75B	Maximum number of servers registered for this device reached
CODM3_E_DEVICE_ NOTSUBSCRIBED	0x8004C75C	Process is not registered as a server on this device
CODM3_E_DEVICE_ NO_MESSAGE	0x8004C75D	No message available
CODM3_E_TRANSFER TIMEOUT	0x8004C760	Message transfer timeout
CODM3_E_MESSAGE_ INSERVICE	0x8004C761	Message in service

Table 64: ODM Error Codes - General ODM Error Codes

© Hilscher, 2009-2010

7.4.2 General ODM Driver Error Codes

Symbol	Value	Description
CODM3_E_DRV_OPEN_ DEVICE	0x8004C7A0	Packet type unsupported by driver
CODM3_E_DRV_INVALID_ IDENTIFIER	0x8004C7A1	Invalid device identifier
CODM3_E_DRV_DEVICE_ PARAMETERS_MISMATCH	0x8004C7A3	Parameters differ from requested device
CODM3_E_DRV_BROWSE_ NO_DEVICES	0x8004C7A4	No devices found
CODM3_E_DRV_CREATE_ DEVICE_INST	0x8004C7A5	Device instance could not be created
CODM3_E_DRV_DEVICE_ NOMORE_TX	0x8004C7A6	Device connection limit reached
CODM3_E_DRV_DEVICE_ DUPLICATE_TX	0x8004C7A7	Duplicate transmitter ID
CODM3_E_DRV_DEVICE_ NOT_CONFIGURED	0x8004C7A8	Device is not configured
CODM3_E_DRV_DEVICE_ COMMUNICATION	0x8004C7A9	Device communication error
CODM3_E_DRV_DEVICE_ NO_MESSAGE	0x8004C7AA	No message available
CODM3_E_DRV_DEVICE_ NOT_READY	0x8004C7AB	Device not ready
CODM3_E_DRV_INVALID CONFIGURATION	0x8004C7AC	Invalid driver configuration
CODM3_E_DRV_DLINVALID MODE	0x8004C7C0	Invalid download mode
CODM3_E_DRV_DLIN PROGRESS	0x8004C7C1	Download is active
CODM3_E_DRV_ULIN PROGRESS	0x8004C7C2	Upload is active

Table 65: ODM Error Codes - General ODM Driver Error Codes

© Hilscher, 2009-2010

7.4.3 ODM cifX Driver Specific Errors

ID of the Error Code	Value	DESCRIPTION
DRV_E_BOARD_NOT_ INITIALIZED	0x8004C001	DRIVER Board not initialized
DRV_E_INIT_STATE_ ERROR	0x8004C002	DRIVER Error in internal init state
DRV_E_READ_STATE_ ERROR	0x8004C003	DRIVER Error in internal read state
DRV_E_CMD_ACTIVE	0x8004C004	DRIVER Command on this channel is activ
DRV_E_PARAMETER_ UNKNOWN	0x8004C005	DRIVER Unknown parameter in function
DRV_E_WRONG_DRIVER_ VERSION	0x8004C006	DRIVER Version is incompatible with DLL
DRV_E_PCI_SET_CONFIG_ MODE	0x8004C007	DRIVER Error during PCI set configuration mode
DRV_E_PCI_READ_DPM_ LENGTH	0x8004C008	DRIVER Could not read PCI dual port memory length
DRV_E_PCI_SET_RUN_ MODE	0x8004C009	DRIVER Error during PCI set run mode
DRV_E_DEV_DPM_ACCESS_ ERROR	0x8004C00A	DEVICE Dual port ram not accessable(board not found)
DRV_E_DEV_NOT_ READY	0x8004C00B	DEVICE Not ready (ready flag failed)
DRV_E_DEV_NOT_ RUNNING	0x8004C00C	DEVICE Not running (running flag failed)
DRV_E_DEV_WATCHDOG_ FAILED	0x8004C00D	DEVICE Watchdog test failed
DRV_E_DEV_OS_VERSION_ ERROR	0x8004C00E	DEVICE Signals wrong OS version
DRV_E_DEV_SYSERR	0x8004C00F	DEVICE Error in dual port flags
DRV_E_DEV_MAILBOX_ FULL	0x8004C010	DEVICE Send mailbox is full
DRV_E_DEV_PUT_ TIMEOUT	0x8004C011	DEVICE PutMessage timeout
DRV_E_DEV_GET_ TIMEOUT	0x8004C012	DEVICE GetMessage timeout
DRV_E_DEV_GET_ NO_MESSAGE	0x8004C013	DEVICE No message available
DRV_E_DEV_RESET_ TIMEOUT	0x8004C014	DEVICE RESET command timeout
DRV_E_DEV_NO_COM_ FLAG	0x8004C015	DEVICE COM-flag not set. Check if Bus is running
DRV_E_DEV_EXCHANGE_ FAILED	0x8004C016	DEVICE I/O data exchange failed
DRV_E_DEV_EXCHANGE_ TIMEOUT	0x8004C017	DEVICE I/O data exchange timeout
DRV_E_DEV_COM_MODE_ UNKNOWN	0x8004C018	DEVICE I/O data mode unknown
DRV_E_DEV_FUNCTION_ FAILED	0x8004C019	DEVICE Function call failed
DRV_E_DEV_DPMSIZE_ MISMATCH	0x8004C01A	DEVICE DPM size differs from configuration
DRV_E_DEV_STATE_ MODE_UNKNOWN	0x8004C01B	DEVICE State mode unknown
DRV_E_DEV_HW_ PORT_IS_USED	0x8004C01C	DEVICE Output port already in use
DRV_E_USR_OPEN_ ERROR	0x8004C01E	USER Driver not opened (device driver not loaded)
DRV_E_USR_INIT_DRV_ ERROR	0x8004C01F	USER Can't connect to device

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
DRV_E_USR_NOT_ INITIALIZED	0x8004C020	USER Board not initialized (DevInitBoard not called)
DRV_E_USR_COMM_ ERR	0x8004C021	USER IOCTRL function failed
DRV_E_USR_DEV_ NUMBER_INVALID	0x8004C022	USER Parameter DeviceNumber invalid
DRV_E_USR_INFO_AREA_ INVALID	0x8004C023	USER Parameter InfoArea unknown
DRV_E_USR_NUMBER_ INVALID	0x8004C024	USER Parameter Number invalid
DRV_E_USR_MODE_ INVALID	0x8004C025	USER Parameter Mode invalid
DRV_E_USR_MSG_BUF_ NULL_PTR	0x8004C026	USER NULL pointer assignment
DRV_E_USR_MSG_BUF_ TOO_SHORT	0x8004C027	USER Message buffer too small
DRV_E_USR_SIZE_ INVALID	0x8004C028	USER Parameter Size invalid
DRV_E_USR_SIZE_ ZERO	0x8004C02A	USER Parameter Size with zero length
DRV_E_USR_SIZE_ TOO_LONG	0x8004C02B	USER Parameter Size too long
DRV_E_USR_DEV_ PTR_NULL	0x8004C02C	USER Device address null pointer
DRV_E_USR_BUF_ PTR_NULL	0x8004C02D	USER Pointer to buffer is a null pointer
DRV_E_USR_SENDSIZE_ TOO_LONG	0x8004C02E	USER Parameter SendSize too large
DRV_E_USR_RECVSIZE_ TOO_LONG	0x8004C02F	USER Parameter ReceiveSize too large
DRV_E_USR_SENDBUF_ PTR_NULL	0x8004C030	USER Pointer to send buffer is a null pointer
DRV_E_USR_RECVBUF_ PTR_NULL	0x8004C031	USER Pointer to receive buffer is a null pointer
DRV_E_DMA_INSUFF_ MEM	0x8004C032	DMA Memory allocation error
DRV_E_DMA_TIMEOUT_CH4	0x8004C033	DMA Read I/O timeout
DRV_E_DMA_TIMEOUT_CH5	0x8004C034	DMA Write I/O timeout
DRV_E_DMA_TIMEOUT_CH6	0x8004C035	DMA PCI transfer timeout
DRV_E_DMA_TIMEOUT_CH7	0x8004C036	DMA Download timeout
DRV_E_DMA_DB_DOWN_FAIL	0x8004C037	DMA Database download failed
DRV_E_DMA_FW_DOWN_FAIL	0x8004C038	DMA Firmware download failed
DRV_E_CLEAR_DB_FAIL	0x8004C039	DMA Clear database on the device failed
DRV_E_DEV_NO_VIRTUA L_MEM	0x8004C03C	DMA USER Virtual memory not available
DRV_E_DEV_UNMAP_ VIRTUAL_MEM	0x8004C03D	DMA USER Unmap virtual memory failed
DRV_E_GENERAL_ERROR	0x8004C046	DRIVER General error
DRV_E_DMA_ERROR	0x8004C047	DRIVER General DMA error
DRV_E_WDG_IO_ERROR	0x8004C048	DRIVER I/O WatchDog failed
DRV_E_WDG_DEV_ ERROR	0x8004C049	DRIVER Device Watchdog failed
DRV_E_USR_DRIVER_ UNKNOWN	0x8004C050	USER Driver unknown
DRV_E_USR_DEVICE_ NAME_INVALID	0x8004C051	USER Device name invalid
DRV_E_USR_DEVICE_ NAME_UKNOWN	0x8004C052	USER Device name unknown

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	DESCRIPTION
DRV_E_USR_DEVICE_ FUNC_NOTIMPL	0x8004C053	USER Device function not implemented
DRV_E_USR_FILE_OPEN _FAILED	0x8004C064	USER File could not be opened
DRV_E_USR_FILE_SIZE_ ZERO	0x8004C065	USER File size zero
DRV_E_USR_FILE_NO_ MEMORY	0x8004C066	USER Not enough memory to load file
DRV_E_USR_FILE_READ_ AILED	0x8004C067	USER File read failed
DRV_E_USR_INVALID_ FILETYPE	0x8004C068	USER File type invalid
DRV_E_USR_FILENAME_ INVALID	0x8004C069	USER Invalid filename
DRV_E_FW_FILE_OPEN_ FAILED	0x8004C06E	USER Firmware file could not be opened
DRV_E_FW_FILE_SIZE_ ZERO	0x8004C06F	USER Not enough memory to load firmware file
DRV_E_FW_FILE_NO_ MEMORY	0x8004C070	USER Not enough memory to load firmware file
DRV_E_FW_FILE_READ_ FAILED	0x8004C071	USER Firmware file read failed
DRV_E_FW_INVALID_ FILETYPE	0x8004C072	USER Firmware file type invalid
DRV_E_FW_FILENAME_ INVALID	0x8004C073	USER Firmware file name not valid
DRV_E_FW_DOWNLOAD_ ERROR	0x8004C074	USER Firmware file download error
DRV_E_FW_FILENAME_NOT_ FOUND	0x8004C075	USER Firmware file not found in the internal table
DRV_E_FW_BOOTLOADER_ ACTIVE	0x8004C076	USER Firmware file BOOTLOADER active
DRV_E_FW_NO_FILE_PATH	0x8004C077	USER Firmware file no file path
DRV_E_CF_FILE_OPEN_ FAILED	0x8004C078	USER Configuration file could not be opend
DRV_E_CF_FILE_SIZE_ ZERO	0x8004C079	USER Configuration file size zero
DRV_E_CF_FILE_NO_ MEMORY	0x8004C07A	USER Not enough memory to load configuration file
DRV_E_CF_FILE_READ_ FAILED	0x8004C07B	USER Configuration file read failed
DRV_E_CF_INVALID_FILETYPE	0x8004C07C	USER Configuration file type invalid
DRV_E_CF_FILENAME_ INVALID	0x8004C07D	USER Configuration file name not valid
DRV_E_CF_DOWNLOAD_ ERROR	0x8004C07E	USER Configuration file download error
DRV_E_CF_FILE_NO_SEGMENT	0x8004C07F	USER No flash segment in the configuration file
DRV_E_CF_DIFFERS_ FROM_DBM	0x8004C080	USER Configuration file differs from database
DRV_E_DBM_SIZE_ZERO	0x8004C083	USER Database size zero
DRV_E_DBM_NO_MEMORY	0x8004C084	USER Not enough memory to upload database
DRV_E_DBM_READ_FAILED	0x8004C085	USER Database read failed
DRV_E_DBM_NO_FLASH_ SEGMENT	0x8004C086	USER Database segment unknown
DEV_E_CF_INVALID_ DESCRIPT_VERSION	0x8004C096	CONFIG Version of the descript table invalid

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	DESCRIPTION
DEV_E_CF_INVALID_INPUT_ OFFSET	0x8004C097	CONFIG Input offset is invalid
DEV_E_CF_NO_INPUT_SIZE	0x8004C098	CONFIG Input size is 0
DEV_E_CF_MISMATCH_ INPUT_SIZE	0x8004C099	CONFIG Input size does not match configuration
DEV_E_CF_INVALID_ OUTPUT_OFFSET	0x8004C09A	CONFIG Invalid output offset
DEV_E_CF_NO_OUTPUT_ SIZE	0x8004C09B	CONFIG Output size is 0
DEV_E_CF_MISMATCH_ OUTPUT_SIZE	0x8004C09C	CONFIG Output size does not match configuration
DEV_E_CF_STN_NOT_ CONFIGURED	0x8004C09D	CONFIG Station not configured
DEV_E_CF_CANNOT_GET_ STN_CONFIG	0x8004C09E	CONFIG Cannot get the Station configuration
DEV_E_CF_MODULE_ DEF_MISSING	0x8004C09F	CONFIG Module definition is missing
DEV_E_CF_MISMATCH_ EMPTY_SLOT	0x8004C0A0	CONFIG Empty slot mismatch
DEV_E_CF_MISMATCH_ INPUT_OFFSET	0x8004C0A1	CONFIG Input offset mismatch
DEV_E_CF_MISMATCH_ OUTPUT_OFFSET	0x8004C0A2	CONFIG Output offset mismatch
DEV_E_CF_MISMATCH_ DATA_TYPE	0x8004C0A3	CONFIG Data type mismatch
DEV_E_CF_MODULE_ DEF_MISSING_NO_SI	0x8004C0A4	CONFIG Module definition is missing,(no Slot/Idx)

Table 66: ODM CifX Driver Specific Errors

© Hilscher, 2009-2010

7.5 ODM DBM V4 Error Codes

ID of the Error Code	Value	Description
CDBM_E_MD5_INVALID	0XC004C810	Checksum invalid
CDBM_E_INTERNALERROR	0XC004C811	Internal Error
CDBM_W_WRITEREGISTRY	0X8004C812	Error writing to the registry
CDBM_E_UNEXPECTED_VALUE _IN_OLD_HEADER_FORMAT	0XC004C813	Error in a file containing the old DBM Header format.
CDBM_E_CHECKSUM_INVALID	0XC004C814	The Checksum of the old Header is invalid
CDBM_E_DB_ALREADY_LOADED _FORMAT	0XC004C815	A database is already loaded
CDBM_E_NO_VALID_TRANSACTION	0XC004C816	No valid transaction handle given
CDBM_E_STD_STRUCT_ ERROR	0XC004C817	An error occured during validation of data
CDBM_E_UNSUPPORTED _DATA_TYPE_ FORMAT	0XC004C818	Unsupported DataType
CDBM_W_CLASS_DELETED _FORMAT	0X8004C819 (Warning)	Using an Object which is marked as deleted
CDBM_W_CLIENT_ DISCONNECTED	0X8004C81A (Warning)	A Client has already an outstanding connection to a Table. The connection is now destroyed.
CDBM_E_STRUCTURE_ DEFINITION_INVALID	0XC004C81B	A structure definition of an Element in a Table is invalid
CDBM_E_NO_DATA_AVAILABLE	0XC004C81C	No data available for this operation
CDBM_E_NO_VALID_STRUCTURE	0XC004C81D	No valid structure available for this operation
CDBM_E_NO_TOGGLE_STRING _FOUND	0XC004C81E	No Toggle string found for this number
CDBM_E_ELEMENT_OUT_ OF_RANGE	0XC004C81F	An element wasn't found in the Record of a Table
CDBM_E_ELEMENT_NOT _IN_TABLE	0XC004C820	The element is not part of the Table
CDBM_E_CANNOT_CONVERT _INTO_CLIENT_TYPE	0XC004C821	The data can't be converted into the Client type
CDBM_E_TRANSACTION_ ALREADY_OPEN	0XC004C822	A transaction is already open. Please close this one first before opening a new one.
CDBM_I_OLD_WITHOUT_ HEADER	0X4004C823 (Informational)	Use of an old DBM file Format without Header
CDBM_E_HR_FROM	0XC004C824	An HRESULT was received from a Subroutine
CDBM_E_PARAMETER	0XC004C825	A Parameter is invalid
CDBM_E_NOTIMPL	0XC004C826	Method is currently not implemented
CDBM_E_OUTOFMEMORY	0XC004C827	Out of memory
CDBM_E_NO_OPEN_ TRANSACTION	0XC004C828	No transaction open
CDBM_E_NO_CONTENTS	0XC004C829	No contents available
CDBM_REC_NO_NOT_ FOUND	0XC004C82A	Record not found
CDBM_STRUCTURE_ELEMENT _NOT_FOUND	0XC004C82B	Element of the Structure not found

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
CDBM_E_NO_MORE_RECORDS _IN_TABTYPE	0XC004C82C	Table type 3 can contain only one record
CDBM_E_WRITE	0XC004C82D	The data in the VARIANT must be given in a SafeArray
CDBM_E_WRITE_NO_ PARRAY	0XC004C82E	The VARIANT contains no valid [parray] element
CDBM_E_WRITE_CANT_ ACCESS_DATA	0XC004C82F	Unable to access SafeArray Data in the VARIANT
CDBM_E_WRITE_DATA	0XC004C830	To write the data of this Element it must be given as a BSTR, or as an Array of VT_UI1/VT_I1
CDBM_E_WRITE_BSTR_E1	0XC004C831	The BSTR string must have an even length.
CDBM_E_WRITE_BSTR_E2	0XC004C832	The BSTR string must contain only hex digits (0..9 and a/A..f/F).
CDBM_E_WRITE_CANT_ INTERPRET_ARRAY	0XC004C833	Unable to interpret data in the SafeArray.
CDBM_E_WRITE_VT_ ERROR	0XC004C834	Data type in the SafeArray is not VT_UI1 or VT_I1.
CDBM_E_WRITE_LENGTH	0XC004C835	Data length is invalid for write operation of this type.
CDBM_WRITE_ELEMENT	0XC004C836	Element not found in the Record of the Table
CDBM_MIN_MAX_ERROR	0XC004C837	Can't write data because of min underflow or max overflow
CDBM_TABLE_EXIST	0XC004C838	Table already exist in the database
CDBM_MIN_MAX_INVALID	0XC004C839	The Min value is greater than the Max Value
CDBM_DEF_MIN_MAX_INVALID	0XC004C83A	The Default Value is not in the range between the Min value and the Max Value
CDBM_CANT_CHANGE_ STRUCTURE_WHILE_ RECORDS_EXIST	0XC004C83B	It's not allowed to change the structure while Records exist in the Table
CDBM_NEW_STRUCT_ NEEDS_TYPE	0XC004C83C	In a newly added structure the data type must be set also
CDBM_VALUE_ERROR	0XC004C83D	Range error while validating a value
CDBM_DATATYPE_UN SUPPORTED_IN_RCS	0XC004C83E	The data type is unsupported in the RCS file format
CDBM_I_COUNT_OF_TABLES _EXCEEDS_ RCS_RANGE	0X4004C83F (Informational)	The count of Tables exceeds the RCS range of Tables. This can cause problems if the file is downloaded to RCS Systems

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
CDBM_I_COUNT_OF_TABLES _EXCEEDS_OLDDBM_RANGE	0X4004C840 (Informational)	The count of Tables exceeds the DBM32.DLL range of Tables. This can cause problems if the file is used with older Tools using the DBM32.DLL
CDBM_UNSUPPORTED_DATATYPE _IN_RCS_MODE	0XC004C841	The Data type is not compatible with the old database format
CDBM_WRITE_UNSTRUC TURED_1	0XC004C842	The data of an unstructured record can only be written with the 'Write' Method not with 'WriteElement'.
CDBM_READ_UNSTRUC TURED_1	0XC004C843	The data of an unstructured record can only be read with the 'Read' Method not with 'ReadElement'
CDBM_WRITE_DATA_LENGTH _INVALID	0XC004C844	The given data length doesn't correspond with the expected data length.
CDBM_UNKNOWN_VIEW_ MODE	0XC004C845	The View Mode is unknown.
CDBM_E_DIAG_TABLE	0XC004C846	It doesn't make much sense to add or delete records from a diagnostic table because those changes are never saved.
CDBM_E_ADR_STRING _ERROR	0XC004C847	The given Address string doesn't fit the required format of this type where all address bytes must be in the range between 0 and FF
CDBM_ERROR_FROM_VAR _CHANGE_TYPE	0XC004C848	Function VariantChangeType return an error when trying to convert the Parameter
CDBM_E_MINERROR	0XC004C849	Error while comparing the Value with the lower range
CDBM_E_MAXERROR	0XC004C84A	Error while comparing the Value with the upper range
CDBM_E_RANGE_ERROR	0XC004C84B	Value out of Range
CDBM_E_TABLE_TYPE1	0XC004C84C	Table type 1 doesn't have a unique record length over all records
CDBM_E_TABLE_TYPE3 _ADDREC	0XC004C84D	Table type 3 doesn't allow to insert more than one Record
CDBM_E_TABTYPE1	0XC004C84E	It's not allowed to insert more Records than structure definitions in Table Type 1
CDBM_E_TOGGLE_ NOT_FOUND	0XC004C84F	Could not find the string for this value in the list of valid toggle strings
CDBM_E_TOGGLE_VALUE_ IS_EMPTY_STRING	0XC004C850	The toggle string for this value is empty.
CDBM_VARIANT2BYTEARRAY _ERROR	0XC004C851	Error during conversion of Variant to byte array

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
CDBM_E_SET_ELEM_ PROP_DEPENDENCY	0XC004C852	The Toggle Type needs also the additional string and the additional number entrys in the Method SetElementProperties
CDBM_E_TABTYPE1_REC_ DOESNT_CORRESPOND_ WITH_ELEMENT	0XC004C853	When reading the records of Table type 1 elementwise the record number must correspond with the element number
CDBM_TABTYPE1_NO_DATA_ FOUND_FOR_RECORD	0XC004C854	When reading the records of Table type 1 and structure definitions are present it's assumed that for each structure element a corresponding record must exist
CDBM_E_TABTYPE1_WRITE_ ELEMENT_NE_RECORD	0XC004C855	When writing the records of Table type 1 elementwise and structure definitions are present it's only allowed to write the corresponding element number in each record
CDBM_E_TABTYPE1_WRITE_ ELEMENT_NOT_FOUND	0XC004C856	When writing the records of Table type 1 with an array and structure definitions are present it's assumed that a corresponding element number of this record exist
CDBM_I_TABLE_NAME_ EXCEEDS_RCS_RANGE	0X4004C857 (Informational)	The Table name exceeds the maximum length of RCS compatible Table names
CDBM_W_CUT_STRING	0X8004C858 (Warning)	The string exceeds the maximum length and will be limited to the maximum length
CDBM_I_STRING_TOO_ SHORT	0X4004C859 (Informational)	The string is below the minimum length. The minimum length will be reduced.
CDBM_I_STRING_TOO_ LONG	0X4004C85A (Informational)	The string is exceeding the maximum. The maximum length will be extended.
CDBM_E_STRING_TOO_ SHORT	0XC004C85B (Error)	The string is below the minimum length.
CDBM_E_STRING_TOO_ LONG	0XC004C85C (Error)	The string is exceeding the maximum length
CDBM_E_WRONG_TYPE_ FOR_WRITE	0XC004C85D	Writing on the Element type with the given Data type is not implemented
CDBM_E_NO_APPEND_IN_ STRUCTURED_RECORDS	0XC004C85E	Method IDbmRecord::AppendData is not allowed for structured records
CDBM_E_DATA_ UNAVAILABLE	0XC004C85F	No data available
CDBM_E_CANT_ CONVERT_INTO	0XC004C860	Unable to convert the value into the Element type

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
CDBM_E_DBM_FILE_ OVERFLOW	0XC004C861	You try to write a RCS like database which needs too mutch bytes
CDBM_E_PW_ERROR	0XC004C862	Password not correct
CDBM_E_FILELENGTH_ CORRUPT	0XC004C863	The file length doesn't correspond to the length given in the Header.
CDBM_E_STRUCT_ TYPE	0XC004C864	Error in the file.
CDBM_E_MD5SUM_ INVALID	0XC004C865	MD5 sum invalid
CDBM_E_STRUCT_ LENGTH	0XC004C866	Error in the expected and given structure length at a specific offset in the file.
CDBM_E_APPEND	0XC004C867	Append of data is only allowed if the Record contains only one data field and the field type will support this
CDBM_APPEND_NOT_ SUPPORTED	0XC004C868	Append of Data not supported by this filed type
CDBM_DATA_TYPE_ APPEND_ERROR	0XC004C869	Can't append Data of this type.
CDBM_E_UNSTRUCTURED_ TABLE_DOESNT_SUPPORT _LENGTH	0XC004C86A	A Table without structure information doesn't support a record length
CDBM_E_DISABLED_WHILE_ TRANSACTION_ IS_OPEN	0XC004C86B	The Method is disabled while a transaction is open. Please close this one first and call the Method again.
CDBM_E_UNABLE_TO_CALL_ READ_ON_LINKED_LIST	0XC004C86C	The Method is disabled on a LinkedList type. Please use the IRecordCollection on this type.
CDBM_E_ELEMENT_HAS_ NO_SUBSTRUCTURE	0XC004C86D	An Element from a Table has no substructure
CDBM_STRUCT_ERROR_ FROM_VAR_CHANGE_TYPE	0XC004C86E	Error from calling VariantChangeType
CDBM_E_FOREIGNKEY_DEF	0XC004C86F	The definition of a FOREIGNKEY must contain the name of the related Table in the description and this Table must exist at this time
CDBM_E_FOREIGNKEY_ REF_TAB	0XC004C870	The description of a FOREIGNKEY must refer to a Table of type 'eDbmTableTypeLinkedList'
CDBM_E_KEY	0XC004C871	To create a Record Collection with a KEY it's necessary to have the data type KEY at the first position in all Records of the searched Table
CDBM_E_KEY_TABLE_ TYPE	0XC004C872	This Method needs a Table of type 'eDbmTableTypeLinkedList'
CDBM_DATATYPE_NOT_ IMPLEMENTED	0XC004C873	This data type is currently not implemented

More see next page

© Hilscher, 2009-2010

ID of the Error Code	Value	Description
CDBM_INSERT_POS_ NOT_FOUND	0XC004C874	The position of the Record where the new one should be inserted wasn't found
CDBM_E_INSERT_REC_QI	0XC004C875	Error during insertion of a Record
CDBM_E_TAB_PROP	0XC004C876	Invalid Property in Table
CDBM_E_KEY_NOT_FOUND	0XC004C877	The KEY wasn't found in the Table
CDBM_E_KEY_INVALID	0XC004C878	The KEY is invalid for this operation

Table 67: ODM DBM V4 Error Codes

© Hilscher, 2009-2010

8 Lists

8.1 List of Figures

Figure 1: Dialog Structure of netGateway DTM 12
Figure 2: Navigation Area 13
Figure 3: Status Bar – Status Fields 1 to 6 16
Figure 4: Status Bar Display Examples 16
Figure 5: Navigation Area - Settings 29
Figure 6: Driver Selection List 30
Figure 7: Driver Selection List - recognized or installed Drivers 30
Figure 8: Driver Selection List - Driver/s selected 30
Figure 9: Device Assignment - detected Devices (* The name of the device class is displayed.) – Example for a device with firmware 32
Figure 10: Device Assignment - Selecting the Device 34
Figure 11: Navigation Area - Configuration 35
Figure 12: License Pane 37
Figure 13: Example: ordering E Mail License request 42
Figure 14: Example: order data form netX License Order Form 43
Figure 15: Status information of the Master 48
Figure 16: Status information of the Slave 48
Figure 17: Pane Command Table 68
Figure 18: Examples - Reading Command with FC 3, Writing Command with FC16 72
Figure 19: Examples - Reading Command with FC 3, Writing Command with FC16 72
Figure 20: Access of the Open Modbus/TCP Client to the netTAP and netBRICK Device (1) 73
Figure 21: Access of the Open Modbus/TCP Client to the netTAP and netBRICK Device (2) 74
Figure 22: Pane Command Table 115
Figure 23: Examples - Reading Command with FC 3, Writing Command with FC16 119
Figure 24: Examples - Reading Command with FC 3, Writing Command with FC16 119
Figure 25: Access of the Modbus RTU Master to the netTAP and netBRICK Device (1) 121
Figure 26: Access of the Modbus RTU Master to the netTAP and netBRICK Device (2) 121
Figure 27: Navigation Area - Diagnosis 141
Figure 28: General Diagnosis 142
Figure 29: Firmware Diagnosis (*The name of the Firmware is displayed.) 145

8.2 List of Tables

Table 1: Descriptions Dialog Panes 5
Table 2: General Device Information 13
Table 3: Hide / display Navigation 13
Table 4: Overview Dialog Panes 14
Table 5: General Buttons 15
Table 6: Table Line – Buttons 15
Table 7: Status Bar Icons [1] 16
Table 8: Descriptions Settings 29
Table 9: Driver Selection List Parameters 30
Table 10: Parameters of the Device Assignment 33
Table 11: Descriptions of the Dialog Panes Configuration 35
Table 12: Communication Change of State 49
Table 13: EtherCAT Slave Parameters 55
Table 14: XML File Names for netTAP and netBRICK 56
Table 15: EtherNet/IP Adapter Parameters (Part 1) 60
Table 16: EtherNet/IP Adapter Parameters (Part 2) 61
Table 17: EDS File Names for netTAP and netBRICK 62
Table 18: Open Modbus/TCP Parameters 66
Table 19: Parameter of the Command Table 68
Table 20: Parameter of the Command Table 70

© Hilscher, 2009-2010

Table 21: POWERLINK Controlled Node/Slave Parameters (Part 1) 77
Table 22: POWERLINK Controlled Node/Slave Parameters (Part 2) 78
Table 23: XDD File Names for netTAP and netBRICK 79
Table 24: PROFINET IO-Device Parameters 83
Table 25: GSDML File Names for netTAP and netBRICK 84
Table 26: SERCOS III Slave Parameters 86
Table 27: XML File Names for netTAP and netBRICK 87
Table 28: CANopen Slave Parameters (Part 1) 91
Table 29: CANopen Slave Parameters (Part 2) 92
Table 30: EDS File Names for netTAP and netBRICK 93
Table 31: CC-Link Slave Parameters (Part 1) 96
Table 32: CC-Link Slave Parameters (Part 2) 97
Table 33: CSP File Names for netTAP 98
Table 34: DeviceNet Slave Parameters (Part 1) 102
Table 35: DeviceNet Slave Parameters (Part 2) 103
Table 36: EDS File Names for netTAP and netBRICK 104
Table 37: Parameters - PROFIBUS-DP Slave (Part 1) 108
Table 38: Parameters - PROFIBUS-DP Slave (Part 2) 109
Table 39: GSD File Names for netTAP and netBRICK 110
Table 40: Modbus RTU Parameters 113
Table 41: Parameter of the Command Table 115
Table 42: Parameter of the Command Table 117
Table 43: ASCII Parameter 125
Table 44: Telegram Structure Part 1 127
Table 45: Telegram Structure Part 2 128
Table 46: Parameter Types 128
Table 47: ASCII Character Table 129
Table 48: Size of the Send and Receive Buffer 130
Table 49: Timing Parameters for the Operating ‘Send Only’ Mode 131
Table 50: Timing Parameters for the Operating ‘Receive Only’ Mode 132
Table 51: Timing Parameters for the Operating Mode 'Client Mode' 135
Table 52: Timing Parameters for the Operating Mode 'Server Mode' 136
Table 53: netSCRIPT Settings 139
Table 54: Explanations to the Dialog Pane Signal Configuration 140
Table 55: Descriptions of the Diagnosis Panes 141
Table 56: Indication General Diagnosis 143
Table 57: Parameter General Diagnosis 144
Table 58: Description Table Task Information 145
Table 59: Error Code Regions 150
Table 60: RCX Errors - MID_SYS Task Error Codes (from Operating System) 152
Table 61: RCX Errors - Common Status Codes (from Operating System) 153
Table 62: RCX Errors - Status & Error Codes (from Operating System) 154
Table 63: RCX Errors - Status & Error Codes – Slave State (from Operating System) 154
Table 64: ODM Error Codes - General ODM Error Codes 155
Table 65: ODM Error Codes - General ODM Driver Error Codes 156
Table 66: ODM CifX Driver Specific Errors 160
Table 67: ODM DBM V4 Error Codes 166
Table 67: Settings (D = Displaying, X = Editing, Configuring) 170
Table 68: Configuration (D = Displaying, X = Editing, Configuring) 170

© Hilscher, 2009-2010

9 Glossary

DTM

Device Type Manager

The Device Type Manager (DTM) is a software module with graphical user interface for the configuration and/or for diagnosis of devices.

FDT

Field Device Tool

FDT specifies an interface, in order to be able to use DTM (Device Type Manager) in different applications of different manufacturers.

© Hilscher, 2009-2010

10 Appendix

10.1 User Rights

User-rights are set within the FDT-container. Depending on the level the configuration is accessible by the user or read-only.

To access the **Settings**, **Configuration** and **Diagnosis** panes of the netGateway DTM you do not need special user rights. Also all users can select the decimal or hexadecimal Display mode or sort table entries.

Note: To edit, set or configure the parameters of the **Settings** and **Configuration** panes, you need user rights for *Maintenance,* for *Planning Engineer* or for *Administrator*.

The following tables give an overview of the user right groups and which user rights you need to configure the single parameters.

10.1.1 Settings

	Observer	Operator	Maintenance	Planning Engineer	Adminis-trator
Driver	D	D	X	X	X
Selecting the Driver	-	-	X	X	X
Device Assignment	D	D	X	X	X
Scanning for Devices	-	-	X	X	X
Selecting the Device	-	-	X	X	X

Table 68: Settings (D = Displaying, X = Editing, Configuring)

10.1.2 Configuration

	Observer	Operator	Maintenance	Planning Engineer	Adminis-trator
Settings	D	D	X	X	X
Signal Mapping	D	D	X	X	X
Memory Card Management	D	D	X	X	X

Table 69: Configuration (D = Displaying, X = Editing, Configuring)

© Hilscher, 2009-2010

10.2 References

[1] Device Type Manager (DTM) Style Guide, Version 1.0 ; FDT-JIG - Order No. <0001-0008-000>

© Hilscher, 2009-2010

10.3 Contacts

Headquarters

Germany
Hilscher Gesellschaft für
Systemautomation mbH
Rheinstrasse 15
65795 Hattersheim
Phone: +49 (0) 6190 9907-0
Fax: +49 (0) 6190 9907-50
E-Mail: info@hilscher.com

Support
Phone: +49 (0) 6190 9907-99
E-Mail: de.support@hilscher.com

Subsidiaries

China
Hilscher Ges.f.Systemaut. mbH
Shanghai Representative Office
200010 Shanghai
Phone: +86 (0) 21-6355-5161
E-Mail: info@hilscher.cn

Support
Phone: +86 (0) 21-6355-5161
E-Mail: cn.support@hilscher.com

France
Hilscher France S.a.r.l.
69500 Bron
Phone: +33 (0) 4 72 37 98 40
E-Mail: info@hilscher.fr

Support
Phone: +33 (0) 4 72 37 98 40
E-Mail: fr.support@hilscher.com

India
Hilscher India Pvt. Ltd.
New Delhi - 110 025
Phone: +91 11 40515640
E-Mail: info@hilscher.in

Italy
Hilscher Italia srl
20090 Vimodrone (MI)
Phone: +39 02 25007068
E-Mail: info@hilscher.it

Support
Phone: +39/02 25007068
E-Mail: it.support@hilscher.com

Japan
Hilscher Japan KK
Tokyo, 160-0022
Phone: +81 (0) 3-5362-0521
E-Mail: info@hilscher.jp

Support
Phone: +81 (0) 3-5362-0521
E-Mail: jp.support@hilscher.com

Korea
Hilscher Korea Inc.
Suwon-Si, 443-810
Phone: +82-31-204-6190
E-Mail: info@hilscher.kr

Switzerland
Hilscher Swiss GmbH
4500 Solothurn
Phone: +41 (0) 32 623 6633
E-Mail: info@hilscher.ch

Support
Phone: +49 (0) 6190 9907-99
E-Mail: ch.support@hilscher.com

USA
Hilscher North America, Inc.
Lisle, IL 60532
Phone: +1 630-505-5301
E-Mail: info@hilscher.us

Support
Phone: +1 630-505-5301
E-Mail: us.support@hilscher.com

© Hilscher, 2009-2010